PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 4 : G06F 15/46

A1

(11) International Publication Number: WO 88/ 08166

(43) International Publication Date: 20 October 1988 (20.10.88)

(21) International Application Number: PCT/US88/01182

(22) International Filing Date: 11 April 1988 (11.04.88)

(31) Priority Application Number: 037,141

(32) Priority Date: 10 April 1987 (10.04.87)

(33) Priority Country: US

(71) Applicant: THE BOEING COMPANY [US/US]; 7755 East Marginal Way South, Seattle, WA 98124 (US).

(72) Inventors: GUMM, Jack, R., Jr. ; 29913 53rd Place South, Auburn, WA 98001 (US). WARD, William, E. ; 16735 235th Avenue S.E., Issaquah, WA 98027 (US).

(74) Agents: ABRAHAMS, Colin, P. et al.; Ladas & Parry, 3600 Wilshire Boulevard, Suite 1520, Los Angeles, CA 90010 (US).

(81) Designated States: AT (European patent), BE (European patent), CH (European patent), DE (European patent), FR (European patent), GB (European patent), IT (European patent), JP, LU (European patent), NL (European patent), SE (European patent).

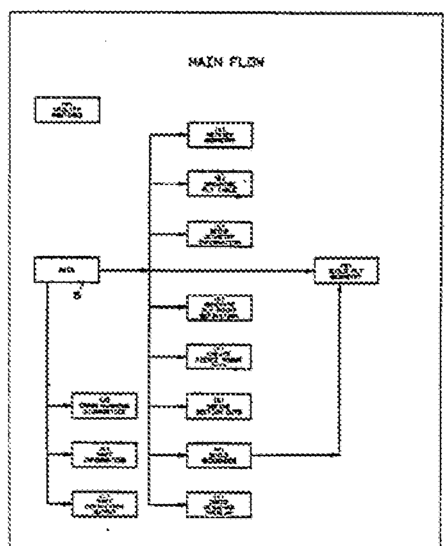
Published

With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: IMPROVED METHOD FOR THE DESIGN AND CONSTRUCTION OF COMPOSITE PARTS

(57) Abstract

An improved system and method are designed to take engineering composite drawings released in a computer-aided design (CAD) data set, determine the attributes of each of the plies into its composite drawing, and pass the information on to using organizations. In preferred embodiments, the system logically determines the geometric definitions for each ply contained in a composite part and reports any errors to the user. Analysis of the physical properties of a part is also provided. Computation of the total part weight and centroid location is performed by the system. Additionally, the present invention creates detailed engineering models of each composite ply and passes information directly to manufacturing.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FR	France	ML	Mali
AU	Australia	GA	Gabon	MR	Mauritania
BB	Barbados	GB	United Kingdom	MW	Malawi
BE	Belgium	HU	Hungary	NL	Netherlands
BG	Bulgaria	IT	Italy	NO	Norway
BJ	Benin	JP	Japan	RO	Romania
BR	Brazil	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SN	Senegal
CH	Switzerland	LI	Liechtenstein	SU	Soviet Union
CM	Cameroon	LK	Sri Lanka	TD	Chad
DE	Germany, Federal Republic of	LU	Luxembourg	TG	Togo
DK	Denmark	MC	Monaco	US	United States of America
FI	Finland	MG	Madagascar		

Description

IMPROVED METHOD FOR THE DESIGN AND CONSTRUCTION OF COMPOSITE PARTS

Technical Field

This invention relates to improved methods and systems for the design and construction of composite parts, and more particularly, to an automated system for integrating the design, analysis and manufacture of composite parts.

Background Art

In certain applications, parts composed primarily of composite materials have significant advantages. A composite airplane wing, for example, can provide twice the life of a conventional metal wing with no increased weight, and, at the same time, provide increased operational capabilities.

Composite parts are fabricated using several layers of composite materials, or plies, that are assembled and cured to form a laminate. The composite material is commonly a fabric or tape that is comprised of fibers having a common orientation. Each layer of this material will have a set fiber orientation along an orientation axis with structural properties that vary in accordance with the relationship to the orientation axis.

The designer of a composite part can combine layers of this material in defined orientations to produce the desired structural properties for the part. Additional layers of material can be added at locations requiring increased strength and the layers can be oriented to maximize resistance in critical load directions.

Thus, each ply of material has several attributes that must be specified by the designer:

(1) the geometric boundary or perimeter shape of the ply must be defined;

(2) the orientation of the fibers for each ply must be defined;

(3) the position of the ply with respect to the other plies in the part (sometimes called the "stacking position") must be defined.

A composite part, such as an airplane wing segment, for example, may be defined by more than 2,000 unique plies.

The design of composite parts is much more complex than designing parts to be fabricated from a homogeneous material, such as metal. Currently, the time required to design, iterate, analyze and program composite parts for fabrication is very high. Because of this complexity, neither human intervention nor manual input can assure that the part will be manufactured as designed.

Disclosure of the Invention

It is an object of the present invention to provide a method and system for automatically generating a sequence book illustrating how to lay down plies during construction of a composite part.

It is another object of this invention to provide a method and system allowing improved structural interrogation of composite parts at various "pierce points."

It is a further object of this invention to provide an improved method and system for automatically feeding data on composite parts into a finite element modeler.

It is another object of this invention to provide an improved method and system for automatically calculating and tabulating the ply weights for individual plies of a composite part and the total weight and centroid of such parts.

It is another object of the invention to provide an improved method and system for reviewing cross sections of composite parts at selected locations and generating corresponding layer and thickness plots.

It is another object of this invention to provide an improved method and system for automatically generating tool paths to cut out the plies of a composite part from sheets of ply material.

It is another object of this invention to provide such an improved method and system which will interface with existing nesting products, such as PINS by Precision Nesting Systems, Inc., of Demarest, New Jersey.

These and other objects of the invention, which will become more apparent as the invention is described more fully below, are obtained by providing an improved system and method. In a preferred embodiment, a System Logic for Integrated Composites (SLIC) system is designed to take engineering composite drawings released in a computer-aided design (CAD) data set, determine the attributes of each of the plies into its composite drawing, and pass the information on to using organizations.

Not only does the information need to be released to all operations organizations, but it also needs to be communicated within engineering groups, such as stress analysis groups. SLIC bridges the gap between engineering, manufacturing, and quality assurance disciplines.

In preferred embodiments, such as SLIC, the system logically determines the geometric definitions for each ply contained in a composite part. If a ply is not completely defined, SLIC reports the error to the user. Automatic error checking assures the designer that the design is of the quality required to meet downstream operations data requirements.

Analysis of the physical properties of a part is also provided to the technical engineering staff. This analysis data may be passed directly from SLIC to NASTRAN, which is an engineering stress analysis system provided by The Mac Neal-Schwendler Corporation of Los Angeles, California. Computation of the total part weight and centroid location is performed by the system.

Additionally, SLIC creates detailed engineering models of each composite ply, based upon the complete composite design. SLIC interprets the design and passes it directly to manufacturing, thus automating the design/build process. Because SLIC forces the designer to correct all geometric definitions, there is a resultant savings in time and money because of reduced errors passed from engineering to manufacturing.

SLIC generates an enormous amount of valuable data that can save thousands of person-hours on a design and build project. Examples of useful data include pierce point data that presents the thickness and properties of the part at any specific location; creation of a table that defines the area, weight and centroid of each ply; a sequence book that supplies the geometry and the order in which the plies are to be placed on a cutting tool; a section view at any location on the part; and point-to-point information to drive numerical control machines in the factory. This information can be used to drive fabric cutters, such as a Gerber Knife, laser inspection machines, and other tools.

Brief Description of the Drawings

Figure 1 is a flow diagram illustrating the operation of the routines within the SLIC system.

Figure 2 is a flow diagram illustrating the operation of the Retrieve Geometry routine.

Figure 3 is a flow diagram illustrating the operation of the Organize Ply Table routine.

Figure 4 is a flow diagram illustrating the operation of the Set Up Geometry Information routine.

Figure 5A is a flow diagram illustrating the operation of the Build Ply Geometry routine.

Figure 5B is an illustration of several unique ply shapes on a common drawing.

Figure 5C is an illustration of the plies of Figure 5B, as determined by visual interrogation.

Figure 5D illustrates the plies of Figure 5B with ply 1 highlighted.

Figure 5E is an illustration of Figure 5E after step 1 of the Build Ply logic has been applied.

Figure 5F is an illustration of Figure 5E after step 2 of the logic has been applied.

Figure 5G is an illustration of Figure 5F after step 3 of the logic has been applied.

Figure 6 is a flow chart illustrating the operation of the Generate Ply Point Definition routine.

Figure 7 is a flow chart illustrating the operation of the Create Pierce Point Definition routine.

Figure 8 is a flow chart illustrating the operation of the Define Section Cuts routine.

Figure 9 is a flow chart illustrating the operation of the Build Sequences routine.

Figure 10 is a flow chart illustrating the operation of the Sequence Overlap routine.

Figure 11 is a flow chart illustrating the operation of the Error Diagnostic routine.

Figure 12 is a flow chart illustrating the operation of the Part Information routine.

Figure 13 is a flow chart illustrating the operation of the Part Definition Output routine.

Figure 14 is an isometric view illustrating a typical ply, layup, and sequence.

Figure 15A is a drawing illustrating several sequences.

Figure 15B is sample composite drawing.

Figure 16 is an illustration of arrowheads on elements in a composite drawing.

Figure 17 is an illustration of relief cuts in a layup.

Figure 18 is an illustration of duplicate elements, overlapping elements, and gaps between elements.

Figure 19 is an illustration of a ply table from a CADAM Standard Library.

Figure 20 is an illustration of a material definition sheet.

Figures 21-23 are illustrations of chain logic in the SLIC system.

Figure 24 illustrates a desired ply shape.

Figure 25 illustrates a method of defining that ply shape using arrowheads and text callout.

Figure 26 illustrates an alternate method to produce the desired shape of Figure 24.

Figures 27-31 illustrate the logical operations to define ply elements using SLIC.

Figure 32 illustrates a multi-sheet drawing of a ply.

Figure 33 illustrates another multi-sheet drawing of a ply.

Figure 34 illustrates several flat patterns for plies.

Figure 35 illustrates an error model display for the SLIC system.

Figure 36 illustrates the screen display for an arrowhead error table.

Figure 37 illustrates a ply table error display.

Figure 38 illustrates the screen display for a layup.

Figure 39 illustrates the screen display for sequence overlap errors.

Figure 40 illustrates the screen display for an information model.

Figure 41 illustrates a finished model part after the operation of SLIC.

Figure 42 illustrates an interactive point model.

Figure 43 illustrates screen displays when using a pierce model in the SLIC system.

Figure 44 illustrates a screen display for pierce points.

Figure 45 illustrates a pierce point coordinate system printout.

Figure 46 illustrates proper drop job control language.

Figure 47 illustrates the program output for section cuts.

Figure 48 illustrates a sequence picture page.

Figure 49 illustrates a prepared sequence format model.

Figure 50 illustrates sequence pages for SLIC sequence output.

Figure 51 is a flow chart illustrating the SLIC system as it interfaces with other systems.

Best Mode for Carrying Out the Invention

A preferred embodiment of the present invention is implemented on an IBM 4381 Series computer using the MVS.XA Operating System in conjunction with CADAM Version II, Release 20.0, and above by means of a software system known as "System Logic for Integrated Composites," or "SLIC." An object code listing for SLIC is provided in Table I of this application. The CADAM system is available from Cadam, Inc., of Burbank, California. The SLIC system also calls routines from the "BCS LIB - Math/Stat/Utility Subprogram Library," available from Boeing Computer Services of Bellevue, Washington. SLIC calls the following routines (and any supporting routines) from BCS LIB: ISRCHV; ISRCH; KOMPRV; HGTIME; HGDATE; HQRWZERO; PROOT; HDGELE; MWGETP; and CHARFL.

The interface of SLIC with existing systems is illustrated in Figure 51. SLIC 1 takes data from the CADAM Engineering Work File 2 and checks it for accuracy. Data from SLIC can be passed on to NASTRAN 3 for finite element analysis of composite parts. Composite part models 4 that have been released by engineering or other design groups are taken by SLIC and used to generate additional information, as discussed in more detail below. SLIC will calculate part information which is stored in a SLIC part data base manager 5, where it can be used by existing nesting systems, such as PINS 6. Data generated by SLIC can interface with a laser, inkjet or other numerical control equipment 7.

The overall design and operation of SLIC is illustrated in the flow chart of Figure 1. Function blocks of SLIC have been labeled A-M, with corresponding flow charts for each block illustrated in Figures 2-13. The Main routine 8 is the main routine for the entire SLIC system. This routine calls all of the driver routines described below. The total area and centroid of a part is calculated in the Main routine.

A. RETRIEVE GEOMETRY (Figure 2)

These routines are used to retrieve and store geometry and text information. This information is formatted and used downstream by the various SLIC routines to output the required data. The information is stored in FORTRAN arrays.

Routine RESOLV (10). SLIC interacts with the CADAM model retrieving and storing information. An understanding of this process can be found in the CADAM Geometry Interface Installation Guide. RESOLV contains several FORTRAN entry points which receive information, such as geometric and text data, from the CADAM model. This information is stored in several arrays to be used by SLIC.

Routine SMXMN (12). This routine calculates the maximum and minimum XY values of a CADAM spline. The maximum and minimum values for each bay in the spline are also calculated. This information is used downstream by the break routines to determine if an element's end points break the spline.

Routine BTXT (14). This routine analyzes text from the ply table, breaks it up, and places it in the proper arrays. The array information will be used in other routines to determine the proper format of the ply table.

Routine SPLTXT (16). This routine splits up the text passed to it. If a dash (-) is found between two values, the range of numbers will be generated (i.e., L1-L4 will generate 1, 2, 3, 4). The numbers generated will be placed in a return array.

Routine AROTXT (18). This routine breaks up the text found on CADAM arrow heads. Arrowheads are used to define the logical path of the geometry on the engineering drawing. Arrowheads are also used to define pierce points. This routine will determine what kind of information has been found and place it in the proper arrays.

Routine FTINFO (20). This routine analyzes text in the sequence format model and stores the data in the proper arrays. This information will be used in the routines that build sequence pages. An explanation of the kind of text found in the sequence format models can be found in the Using SLIC section below.

Routine TXTNFO (22). This routine analyzes and breaks up the attribute text found on the section cut lines. An explanation of section cuts can be found in the Using SLIC section below.

(B) ORGANIZE PLY TABLE (Figure 3)

These routines will organize the standard ply table information in a format to be used in SLIC. If the information has been improperly defined, error messages will be generated.

Routine BLDTBL (24). This routine is the driver routine for formatting or merging all of the information into standard ply tables. The location of notes is critical to the building of the table. Each dash number/page number note will determine the grouping for each ply table

page. If a ply table information note is placed too far away from a dash number/page number note an ERROR message will be generated. This distance can vary depending on the width of the ply table page desired. A sample picture and a brief explanation of the ply table can be found in the Using SLIC section, below. The Users' Manual also defines the ERROR messages that may be generated by these routines.

The ply table is important to SLIC because it defines how the composite part is to be built on the tool.

Ply Table Note Definitions.

The ply table notes for SLIC have been placed in a separate CADAM view to avoid confusion with notes in other views.

Ply Table: - 101 page 1

Ply Table: - 102 opposite page 1

This note is used to define each ply table page. The word "opposite" used in the second note shown tells SLIC that an opposite hand part is being defined.

S20. This note is used to define each sequence number.

L10. This note is used to define each layup number.

45. This note is used to define orientation associated with a layup number. The value of the orientation must be followed by a degree symbol.

7250. This note is used to define the material code. This note must be four numberic characters.

Routine NFOUND (26). This routine will determine if no ply chart information was found in the CADAM models flagged for the SLIC run. For example, if sequence numbers are not defined or placed in the proper view, an ERROR message will be generated.

Routine MERGE (28). This routine will merge the sequence numbers, orientations, and material codes into all

the ply table pages. Once the information is merged it will be sorted in Y decending order.

Routine MERGE1 (30). This routine will merge the layup numbers into all the ply table pages. Once the information is merged it will be sorted in Y Y descending order.

Routine MATMRG (32). This routine will match or merge the material codes with the proper layup number for each ply table page.

Routine SEQMRG (34). This routine will match or merge the sequence numbers with the proper layup number for each ply table page.

Routine GRNMRG (36). This routine will match or merge the orientations with the proper layup number for each ply table page.

Routine VDASH (38). This routine will verify that dash number information has been defined properly. For example, page 1 may have been used more than once for a given dash number. ERROR messages will be generated if errors are found.

Routine VSEQU (40). This routine will verify the sequence information to make sure valid values have been obtained.

If a duplicate sequence of numbers for a dash number is found, an ERROR message will be generated.

Routine VLAYUP (42). This routine will verify the layup information to make sure valid values have been obtained. If a duplicate layup for a dash is found, a WARNING message will be generated. If duplicate layup is found for a dash with different material and/or rotations, an ERROR message will be generated. If a layup number is

defined in the same sequence, an ERROR message will also be generated.

Routine MCHECK (44). This routine checks the material code defined to see if that material has been defined to SLIC. The actual material code properties are defined in a routine called DEFINE.

If a material is not defined in SLIC, then a message will be generated.

(C) SET UP GEOMETRY INFORMATION (Figure 4)

The information received from the RETRIEVE routines (Section (A)) is put in the arrays in a random fashion. These routines organize the arrays and set up additional arrays so that processing of the data can be accomplished by other sections.

Routines CCHAIN (46), BCHAIN (48), BCHAN1 (50). These routines will order the array that contains the elements into groups of chained sections. A pointer array is set up that points to each section of chained elements. A section is considered chained when the end points chain to more than one element or do not chain to any elements.

Routine VERIFY (52). This routine will look at all the arrow heads and find the element that the arrow head lies on. Then the chained section the arrowhead belongs to is identified and stored in an array. This routine assumes that the elements have been chained into sections by the chain routines. It also assumes the array containing the arrowheads has been sorted so all the arrow-heads that are in the same location are next to each other in the arrowhead array.

Routine BREAK1 (54), BREAK2 (56). These routines look through all the elements in the element array and

determine if an element is a breaker element. If it is, a flag is placed on it. A breaker element is defined as an element that has one of its endpoints that lies on another element. This routine is only called once. By looping through all the elements once and placing flags on them, the process of breaking elements in the BLDPLY logic section (D) is greatly reduced. This is because all the elements don't need to be checked for breaking each time, only the ones with flags placed on them by these routines.

Routines DUPTBL (58), DUPARY (60), PUTMSG (62), RSAME (64). These routines set up an array that points to the section of arrowheads that define a ply. When the main routine builds a specific ply, this array is searched by routine DUPARY and all plys that have arrowheads that point to the same sections are determined to have the same geometry. This saves reprocessing the plies that are identical.

(D) BUILD PLY GEOMETRY (Figure 5)

These routines are called to build the geometry definitions for each of the plies. When engineering drawings are designed on a computer, the mathematical definitions for the plies can be extracted automatically and used for processes that need the definitions, such as planning, quality, numerical control (NC), programming, and engineering organizations like weights and the technical staff. The problem then is to identify all and only the elements that define the ply boundaries.

Build Ply Geometry operates using a process that, given the original input of geometry and arrowheads placed at logical positions so a person can visually determine where a ply is defined, a ply definition of elements limited to the actual edge of the closed boundary will result, thus producing a definition that can be used for all the disciplines requiring the ply definition. The

example is a simple example to illustrate the concept, but the process will work on very complex ply definitions.

Figure 5B defines three unique ply shapes. Figure 5C shows the ply definitions that would result from visual interrogation. The process for this invention will now be predented to illustrate the same results for ply 1 (Figure 5D):

Step 1. The first step will identify each arrowhead for the ply with an element. Then those elements are chained together on each end point until the end point does not chain or chains to more than one element. This will result in the elements shown in Figure 5E.

Step 2. Every element that is broken by the endpoint of another is then separated into two elements. For example: the vertical line will be broken (or divided) into two lines (Figure 5F).

Step 3. The last step is almost identical to step 1 except because of the elements being broken the result will be a group of elements only defining ply 1 (Figure 5G).

Several of the routines in the section are needed only to handle the arrays that are used in SLIC. The logic could be executed in various ways. The main routines for this section are BLDDRV, BUILD1, BREAK, BUILD2.

<u>Routine BLDDRV</u> (66). This routine is the main driver for executing the build ply logic. The ply number that is to be built is passed in and an array of all the elements that make up the ply boundaries is returned. This routine assumes all the arrays have been set up previously by the Set Up Geometry routines described in section (C) above.

<u>Routine BUILD1</u> (68). This routine executes the first step of the build ply logic. All arrowheads for the specific ply are identified and connected to one of the elements in the model. Then the connecting elements to

each end of the arrowhead elements are found until a branch or no chain is found. All these elements are placed in a table with pointers and flags to keep track of where they came from and what chained section they belong to. (See decription of Step 1, Figure 5E, above).

Routine BREAK (70). This routine finds the elements that are to be broken and breaks the element by creating two elements in the table instead of one. The routine BREAK calls other routines that handle pointers and flags on the new elements. (See description of step 2, Figure 5F, above).

Routine UPDTE1 (72), UPDTE2 (74), SPLIT (76), RANGE (78). These routines are used for handling pointers and flags on elements. They are specific to the methods used to store the elements in the arrays and could be dependent on the various data array structures.

Routine BUILD2 (80). This routine looks at the new sections that have been created by breaking elements. It determines what sections still have arrowheads on them and builds the final array of the elements making up the ply definition. It then passes the array back to the calling routine. (See description of step 3, Figure 5G, above).

(E) GENERATE PLY POINT DEFINITION (Figure 6)

This section of routines will generate strings of XY points that will define the shape of each layup or ply within a part. These points can be used for several applications, such as tracing or cutting the part on a numerical control machine.

Routine CANDO (82). Routine CANDO is the main driver for generating the GOTO points for defining each

layup or ply. This routine assumes the geometry for defining the plies has already been defined.

Routine CNDCHN (84). This is the driver routine for routine CHAIN1. This routine will group all profile geometry together. If all geometry does not chain, the element end points where the chaining error is found will be tagged with a note.

At this point, the closed shapes have been grouped or chained together in their element form (lines, arcs, splines).

Routine CHAIN1 (86). Routine CHAIN1 will chain the geometry by its end points. The array will be sorted in the order the geometry is chained. The first element to be chained should be at the top of the array. The array will be processed until the beginning point in the array is found again or until the end of the array is reached. If the end of the array is found before all the geometry chains, an ERROR message will be generated.

Routine POINTS (88). This routine will break up the individual elements (lines, arcs, splines) into GOTO points. The GOTO points will be generated according to their chained order. The number of GOTO points generated is dependent on the cord height tolerance defined. Arrays will be set up to point to each set of GOTO points that define a profile or closed shape. The points stored in a GOTO point table will later be sent to routines to order the points in clockwise or counterclockwise order.

Routine PTLN (90). Routine PTLN put the line end points into the GOTO point array.

Routine PTARC (92). This routine breaks up the arc into GOTO points based upon the cutting cord height tolerance.

Routine PTSPL (94). Routine PTSPL breaks up spline into GOTO points based upon the cutting cord height tolerance.

Routine PROFIL (96). This routine will calculate the XMIN, XMAX, YMIN, YMAX of each of the closed shapes or profiles in the GOTO point table. The pointers to the largest pocket are moved to the beginning of the table that points to each profile. The largest profile is considered the outside profile. Routines will be called to make sure the smaller shapes are completely contained by the outside profile.

Routine PROFLL1 (98). This routine will call pierce for the points on the internal cutouts to determine if all the cutouts are contained within the exterior profile.

Routine PROFL2 (100). This routine will determine if internal cutouts overlap or are inside the other cutout (illegal cutout).

Routine DCWCCW (102). DCWCCW is a driver routine for CWCCW. Each pocket's GOTO points will be passed to routine CWCCW to see if the points are going clockwise or counterclockwise. The outside profile should be ordered clockwise and the internal cutouts should be counter-clockwise. If the internal pockets or outside profile are not ordered properly DCWCCW will reorder the GOTO points in their proper order.

Routine CWCCW (104). CWCCW will analyze an array of XY points to determine if they are in clockwise or counterclockwise order and return the answer.

Routine CNTRD (106). This routine will calculate the AREA1 and the CENTROID for a given set of points.

Formula: ABS (Summation X(I)*Y(I+1) = Y(1)*X(I+1))/2

Formula for CENTROID uses a trapezodial method.

This routine assumes the array of XY points close and the first and last points are the same in the array.

Routine OPTMZE (108). Routine OPTMZE will optimize the GOTO points based upon the starting load point or XLOAD, YLOAD. The routine searches for the closest point in the internal and external profiles and reorders the GOTO points. This will minimize machine time by generating a more efficient tool path. Routine ORDER is called to reorder the GOTO points array.

Routine ORDER (110). This routine is called by OPTMZE to reorder a string of XY GOTO points by passing the starting location to begin reordering.

(F) CREATE PIERCE POINT DEFINITION (Figure 7)

These routines are used to determine all the plies a pierce point penetrates. Calculations of the laminate at the pierce point are also done. An explanation of pierce points found in the "Using SLIC" section below.

Routine PRCDRV (112). This routine will take a string of points defining a ply and determine if a given point lies within the boundary definition or outside of it. It also handles internal cutouts. This routine is actually a driver for routine PIERCE.

Routine PIERCE (114). This routine will determine if a point lies within a closed polygon shape. Routine PRCDRV determines if the polygon is a cutout or the outside boundary.

Routine PCHART (116). This routine is the driver routine for calculating and creating information for each pierce point that was identified by the user.

Routine PSTRES (118). Subroutine PSTRES creates the actual pierce chart or output data for each pierce point. It calls routines for calculating the information.

Routine CONSTR (120). This routine computes the (A), (D), (B), (S) matrices for a laminate given the following information for each ply of material in the laminate: bias or grain direction, material properties, and location within the laminate. The calculation of the properties are based upon laminate plate theory.

Routine NCARD (122). This routine prepares NASTRAN, PSHELL and MAT2 cards for interface to NASTRAN.

Routine MAT2 (124). This routine prints out NASTRAN MAT2 cards.

Routine WRTMAT (126). Subroutine WRTMAT will write out the material arrays. This routine is used to provide a standard output file for the material properties. It is called when a list of the properties are wanted for interface to other programs.

Routine PCTGET (128). This routine will retrieve the array of all the plies a particular pointed pierced. This information is retrieved before PCHART calls PSTRES.

Routine PCHECK (130). This routine will read in all the pierce point data into a pierce point array. This is done before the ply definitions are created. This routine also checks for duplicate point data.

(G) DEFINE SECTION CUTS (Figure 8)

These routines will generate defined cross section cuts through a series of layups comprising a composite part. A brief explanation of section cuts can be found in the "Using SLIC" section below.

Routine SSTICK (131). This routine is the driver routine for taking cross section cuts through the composite part. This routine will loop through each section cut line and then call other routines to generate the cross section data.

Routine STKBLD (132). This routine builds an array of information for each ply that is cut by the section line cut. The distance and order of the cut will be stored.

Routine DLLIM (133). This routine will determine the limits (XY minimums and maximums) of a line given its XY end points.

Routine INTDRV (134). This routine finds all intersecting points for a section cut line, given strings of points defining the layup or ply.

Routine LIMITV (170). This routine checks the limits (XY minimums and maximums) of two closed profiles to see if they overlap. This is the same routine called in the CHECK SEQUENCE OVERLAP logic described in section I below.

Routine INTERC (136). This routine will calculate the intersection point for two lines. The point of intersection is returned. An alternate return will occur if the lines are parallel.

Routines PRCDRV (112) and PIERCE (114). These routines create pierce point data, as explained in the CREATE PIERCE POINT DEFINITION, section F, above.

Routine STKMOD (139). This mode is a driver routine for creating section cut drawings or CADAM models. Stick figures, gage data or thickness plots, and (E x T) models will be generated. This routine will create separate models for each section cut line.

Routine STKCAL (140). This routine will calculate all the information to build a stick figure model. Each line of the stick figure will display the layup number, sequence number, orientation and material code of each layup cut.

Routine CONSTR (120). This routine computes (A), (D), (B), (S) matrices for a laminate given the following information for each ply of material in the laminate: bias or grain direction, material properties, location within the laminate. The calculation of the properties is based upon laminate plate theory.

This routine computes the (A), (D), (B), (S) matrices for a laminate as well as the thermal load vectors AT, BT, DT. An average density PMO is also computed.

```
          (Q11   Q12    0)
Let Q =   (Q21   Q22    0)   where Q11 = E1/(1-NU12*NU21)
          (0     0    Q33)         Q22 = E2/(1-NU12*NU21)
                                   Q12 = NU12*E2/(1-NU12*NU21)
                                   Q66 = G12

                             Note that NU12*E2 = NU21*E1.


          ( C*C    S*S    S*C    )
Let T =   ( S*S    C*C   -S*C    )  where S = Sin(theta)
          (-2SC    2SC  C*C-S*S)             C = Cos(theta)
```

The transformed matrix is then:

Q-BAR = T-TRANSPOSE * Q * T

Let $S = \begin{pmatrix} G13 & 0 \\ 0 & G23 \end{pmatrix}$ and $TS = \begin{pmatrix} C & S \\ -S & C \end{pmatrix}$

The transformed shear matrix is:

S-BAR = TS-TRANSPOSE * S * TS

```
      SUM
(A) = MAT Q-BAR *(Z1 - Z0)            over each layer -
      ION                               bottom to top

      SUM
(B) = MAT Q-BAR *(Z1^2- Z0^2)/2       over each layer -
      ION                               bottom to top

      SUM
(D) = MAT Q-BAR *(Z1^3-Z0^3)/3        over each layer -
      ION                               bottom to top

      SUM
(S) = MAT Q-BAR *(Z1 - Z0)            over each layer -
      ION                               bottom to top

      SUM
PMO = MAT RHO *(Z1 - Z0)              over each layer -
      ION                               bottom to top
```

The average density is then PMO/total thickness.

The thermal expansion coefficients A1 and A2 are transformed to laminate coordinates by:

A-X = C*C*A1 + S*S*A2

A-Y = S*S*A1 + C*C*A2

.5A-XY = 2*S*C*(A1-A2)

Let A-BAR denote the column vector of transformed coefficients. Then the thermal load vectors are computed

by:

$$(AT) = \underset{\text{ION}}{\overset{\text{SUM}}{\text{MAT}}}\ Q\text{-}BAR*A\text{-}BAR\ *(Z1 - Z0)$$ over each layer - bottom to top

$$(BT) = \underset{\text{ION}}{\overset{\text{SUM}}{\text{MAT}}}\ Q\text{-}BAR*A\text{-}BAR\ *(Z1^2 - Z0^2)/2$$ over each layer - bottom to top

$$(DT) = \underset{\text{ION}}{\overset{\text{SUM}}{\text{MAT}}}\ Q\text{-}BAR*A\text{-}BAR\ *(Z1^3 - Z0^3)/3$$ over each layer - bottom to top

<u>Routine STKSTK</u> (141). This routine takes all the information compiled in routine STKCAL and generates the actual stick figure model.

<u>Routine STKGAG</u> (142). This routine will build S gage or thickness plot of the section cut. This will be built in the same model below the stick figure plot.

<u>Routine STKEPL</u> (143). This routine will build the plot for modulos of elasticity (E) and the modules of elasticity times thickness (E x T) along the section cut.

(H) BUILD SEQUENCES (Figure 9)

These routines are used to build a book of all the ply sequences in a part. This book can be used by engineering for part visibility. Planning can also use the book for a planning tool.

<u>Routine BLDSEQ</u> (150). This routine is the main driver for creating sequence drawings. A sequence model for each sequence within a given part/dash number will be generated.

Routine BLDET (152). This routine will generate geometry defining each layup on a separate page for each layup. For sequence models all the layups for a given sequence will be generated and dittoed on the main drawing area.

Routine STMODL (154). This routine will start a model with the proper root name and the sequence number inbedded in the name. The model started will be used by SLIC to write out the geometry and text information for the sequence being built.

Routine RETFMT (156). This routine will call CADET to initialize the model to be used for that sequence. The routine will then look and see if a sequence format model has been created and, if so, load it into model common.

If CADET is called, format notes will be analyzed to see if special notes with the proper values have been placed in the sequence format model. An explanation of the sequence format model and the special notes required can be found in the "Using SLIC" section below.

Additional SLIC information will be placed on the NOSHOW page in view PV for each sequence drawing built.

Routine PSNOTE (158). This routine will place text on the sequence model being created. This text is the page number, which will be incremented for each new page, and the sequence number which is obtained from the standard ply table information. The notes will be placed at the locations defined on the sequence format model by the special notes.

Routine FMTINT (160). This routine will initialize the values for the format notes. If the format notes are not found, these values will not be updated, causing a sequence format model error.

Routine FMTVER (162). This routine will verify if valid values on the special notes in the format models were defined. If invalid notes or values have been defined, an ERROR message will be generated.

An example format model and an explanation of the special notes can be found in the "Using SLIC" section below.

Routine NSCHT (164). This routine will write out SLIC version number, date and time of run, and models flagged to create part. This chart will be placed on the NOSHOW page on every sequence page generated by SLIC. The CADAM group and user in which the models were flagged will also be given. This information will be used as accountability and traceability information.

(I) SEQUENCE OVERLAP (Figure 10)

This segment of routines will look at all the shapes on a given sequence and determine if any of the shapes overlap. This is important to SLIC because the sequence has been improperly defined if overlap occurs.

Routine COVDRV (166). This routine compares two shapes that can have internal cutouts and determines if they have any common overlapping or cover. The shapes are in the form of strings of XY points.

Initially the XY limits of the shapes will be checked to see if they overlap. Routines will be called to determine if the first shape completely covers, partially covers, or does not cover the second shape.

Routine COVER (168). This routine takes two strings of ordered, closed-shaped XY points and determines if they have any common overlap or cover.

Information will be returned to tell how or if the first string overlaps the second string, and how or if the second string overlaps the first.

<u>Routine LIMITV</u> (170). This routine checks the limits (XY minimums and maximums) of two closed profiles to see if they overlap.

<u>Routine HLDRV</u> (172). This routine is the driver routine for routine HIDELN. This routine will compile how the lines in the first profile string relate to the lines in the second profile string. This information will be used down stream by routine COVER.

<u>Routine HIDELN</u> (174). This routine will determine how a line is hidden or covered with respect to a string of points.

<u>Routine INTERC</u> (176). This routine will calculate the intersection point for two lines.

The point of intersection is returned.

An alternate return will occur if the lines are parallel.

<u>Routine OVRLAP</u> (178). This routine will determine how much of the first line is covered by the second line. The percentage of overlap is returned. It is assumed the two lines are parallel. The first check in the routine will determine if the lines are colinear.

(J) ERROR DIAGNOSTICS (Figure 11)

These routines output various error and warning messages to an error file. The actual messages are generated in several routines which will be discussed in the various sections.

Routine PLYCHT (180). This routine will write out the ply chart error messages to the error file. An example of the error file and an explanation of each message can be found in the "Using SLIC" section below.

Routine PLYWRN (182). This routine will write out the ply chart warning messages to the error file. An example of the warning file and an explanation of each message can be found in the "Using SLIC" section below.

Routine FMCHT (184).. This routine is a driver routine for adding sequence format error and warning messages to the error file. An example of the error and warning messages and an explanation of each can be found in the "Using SLIC" section below.

Routine FMERR (186). This routine will write out the format chart error messages to the error file.

Routine FMWRN (188). This routine will write out the format chart warning messages to the error file.

(K) PART INFORMATION (Figure 12)

These routines generate the layup (LYUP) and information (INFO) CADAM models or files. This information describes the part in detail by detailing each ply's weight, area, etc., as well as the geometric information of each ply.

Routine MLCHAR (190). This routine will write out a chart of all the layup pages that are built in each CADAM file. A sample layup chart can be found in the "Using SLIC" section below.

Routine BLDET (192).. This routine will generate geometry defining each layup on a separate page for each

layup. These pages are in a CADAM model or file. The designer can view the layup geometry to make sure that the desired results were obtained.

Routine FCHART (194). Routine FCHART is the driver routine for formatting the part information and calling routine FBUILD to generate a chart to generate part information. This information is found in the CADAM INFO model or file. A sample INFO file and an explanation of its contents can be found in the "Using SLIC" section below.

Routine FBUILD (196). Routine FBUILD will build the information chart for a part number. It is called "FCHART."

(L) PART DEFINITION OUTPUT (Figure 13A)

These routines output a part definition or generic file of all the information about the part. This information can be used to feed to other programs, systems, machines, etc. An explanation of the routines and the generic output will be given.

Routine GENOUT (198). This routine will loop through the part information and generate generic output for all the layups in the part. Opposite hand part XY points will be flipped.

Routine GTFLIP (200). This routine will flip a string of XY points about the Y axis.

(M) UTILITIES ROUTINES (Figure 13B)

These routines are utility routines that are used in several routines in SLIC.

Routine UNIT. This routine will unitize a vector (A, B, C).

Routine UNITD. This subroutine will unitize a vector in double precision. A, B, C = direction cosines.

Routine ICNTER. Returns the number of characters in a character string. The character count is determined by scanning the string from right to left for the first non-blank character.

Routine CONVTI. Converts an integer to a character string.

Routine MVCHRL. This routine uses the IBM "MOVE CHARACTER LONG" (MVCL) routine to transfer data from one storage location to another. The length is limited only by the MVCL instruction.

Routine PROJLN. This routine will determine if a point lies on a line.

Routine PROJAR. This routine will determine if a point lies on an arc.

Routine PROJSP. This routine will determine if a point lies on a spline.

Routine CMPRSS. This routine compresses a text array by eliminating all blanks.

Routine MODSRT. This routine will create the model name for the different models in SLIC. It can be modified by each installation for desired model names.

Routine LIMIT. This is a routine that determines if a point X, Y lies within the defined limits XY maximums and minimums.

Routine DEFINE. This routine defines the material properties for each of the material codes. If a material code is placed in the standard ply table and not defined by this routine, an error message will be generatd by SLIC. An example of the information for each material that is input into SLIC can be found in the "Using SLIC" section below.

Routine GTPRM. This routine will read the parameter or PARM card on the SLIC execute step in the Job ntrol Language (JCL). A list and explanation of each ion will be given.

SLIC PARM OPTIONS

When SLIC is executed, there are several options that can be specified. If no options are specified, SLIC will break apart the layup geometry and build LYUP model(s). This is a good method to use to debug geometry and arrowhead errors. The following are options that can be specified in the PARM field.

CHTOL. CHTOL is the chaining tolerance used to chain geometry and determine if arrowheads are on elements. The tolerance set in SLIC is .004.
Example: Chaining tolerance = .005. PARM = "CHTOL = .005"
The default can also be changed and linked into SLIC in block data.

CUTOL. This parameter is the chord height tolerance used for breaking up arcs and splines into a string of points. the default in SLIC is .01.
Example: Cutting tolerance = .05 PARM = "CUTOL = .05"

This default can also be changed and linked into SLIC in block data.

TABLE. This option will turn on the option for SLIC to read the ply table and create output charts in the INFO model(s). The remainder of the options all require this option to be turned on.
Example: PARM = "TABLE"

PIERCE. This will create pierce data in the PRCE model(s) if there are pierce points called out in the layup model(s).
Example: PARM = "TABLE, PIERCE"

SEQUNC. SEQUNC will produce sequence drawings the dash numbers that have a start sequence format.
Exa ple: PARM = "TABLE, SEQUNC"

PLEFT. This option will cause pierce points to pierce the part in the left-hand coordinate system.
Example: PARM = "TABLE, PIERCE, PLEFT"

BPERCE. This will read pierce points from an input file and create NASTRAN cards for each of the pierce points. For an example of the input format, see the "Using SLIC" section below.
Example: PARM = "TABLE, BPERCE"

GENERIC. If this option is turned on, generic output will be generated. The generic output defines all the layups for the part. The output for the layups is unformatted, variable-length records.
Example: PARM = "TABLE, GENERIC"
The DD card that must be allocated with this option is FT12F001, with the data set being a variable block, unformatted record.

```
Example: FT12F001 DD DSN+&&GENERIC,DISP+(,PASS),VOL=SER=,
                  UNIT=3380,SPACE=(TRK,(5,5),
                  DCB=(RECFM=VBS,LRECL=32752,BLKSIZE=32756)
```

The following is an explanation of the generic output and the read statements required to obtain the information for each layup.

GENERIC OUTPUT

```
LOGICAL *1 PART (20)
REAL *4     ROT(8),XLPT(2),CENTRD(2),RLIM(4),POINTS(2,1000)
INTEGER     IDASH,ISEQ,LAYUP,MCODE,IROT,NPTS
DIMENSION   EMPNO(2),DATE(2),FMODEL(20,20)
READ (XX,END=1000)IFMOD, ((FMODEL(I,J),I=1,20),J=1,IFMOD),
*   IDASH,ISEQ,LAYUP,MCODE,IROT,
*   (ROT(I),I=1,IROT),AREA,WEIGHT,VOLUME,
*   (XLPT(I),I=1,2),(CENTRD(I),I=1,2),GRUP,
    (EMPNO(I),I=1,2),(DATE(I)=1,2),VERSN,NPOC,NDLAY)
    DO 100 K = 1, NPOC
            READ (XX) ((RLIM(I),I=1,4),NPTS)
            READ (XX) ((POINTS(J,I,J=1,2),I=1,NPTS))
    100 CONTINUE
```

The profile for the layup is output first, followed by the cutouts.

All layups are in sequence order for the part.

USING SLIC

INTRODUCTION

This section provides the requirements and information to access and operate SLIC. An overview of the purpose of SLIC is given as well as an explanation of its operation. The requirements and logic used are explained and outlined. A method for executing SLIC is presented.

SLIC is an advanced application program for use by operators with training and experience in using CADAM systems.

OVERVIEW

Software Logic for Integrated Composites (SLIC) is a program written specifically for use with flat developed part definitions for composites. It is executed against models generated from the CADAM Graphics System. The program has access to CADAM models by use of CADCD and CADET, which are modules of the geometry interface package of CADAM.

SLIC reads a CADAM model or set of models and creates additional models that contain complete definitions for each part or layup separate from all other parts or layups. The program uses a function of CADAM called "detail pages." A new detail page is created for each indvidual part or layup geometry. CADAM has a limited amount of detail pages per model and a limited model size. Because of this, an additional model will be created when the SLIC model reaches 50 details or approximately 10K of model size. An error model is also created that contains diagnostics if arrowheads used for geometry definitions are not placed on elements. The error model also contains ply table errors, and a warning chart for possible errors.

Sequence of layup is controlled by a ply table. This table has been structured to allow rapid revision of the table and requires only that the sequences be numerically in order. The table itself may be out of order. This table will permit automatic generation and plotting of sequence drawings.

SLIC also permits pierce of the geometry at any interactively defined point(s). Through batch, any large number of points may be transmitted to a data set for printed output or interface with analysis programs such as NASTRAN. The pierce output data is described further in this document.

Upon satisfactory completion of a model which passes the SLIC checks, data can be passed to a SLIC Part Data Manager which permits the planning department to execute special manufacturing routines. These routines

will interface with nesting software to nest parts with common materials and output centerline data for automatic knife cutting of composites.

DEFINITIONS

"Arrowhead text" means a specific CADAM element that allows text to be attached to an arrowhead.

"Chaining tolerance" means the tolerance between the end points of elements. SLIC uses a tolerance of +.004. This tolerance can be easily modified.

"Elements" means the geometric entities on CADAM, such as lines, circles, ellipses, and splines.

A "layup" may consist of 1 to 8 plies, provided they have the same geometry, same material, and are mated. Grain direction of the plies which make up the layup may vary, but must be identified in the ply table. If the thickness of raw material, type of material, or geometry is not the same, the layup number must be different. If any other layup resides between geometrically and materially identical layups, these may be assigned the same layup identity, provided they reside in different sequences. The "layup" definition is shown in Figure 14.

"Ply" means an individual piece of material. Several plies may be laid together to make a layup and/or a part.

"Sequence" may be synonymous with "level." An example of a sequence with one or more layups is shown in Figure 15B.

"Ply tables" are shown and clarified in Figure 19.

INPUT REQUIREMENTS

CADAM Geometry Identification Rules

For the purpose of explanation and instruction of SLIC, a small composite is used as an example. The example design drawing is shown in Figure 15A.

Note that the View PV axis corresponds with the desired 0° (3 o'clock) position of ply orientation. Positive angles are counterclockwise, and negative angles are clockwise from the 3 o'clock position. This correlation of the PV axis is critical for downstream output.

Define Layups in View PV

All layups must be defined in CADAM View PV. All other views are ignored for geometry. View PV was chosen as the view to use because SLIC presently deals only with 2-D projections or flat patterns of parts. For the merging of several models, the common view selected by SLIC is View PV.

PV Views Must Coordinate

SLIC can be executed on several models at a time. When more than one model is selected, the PV views in the models are overlaid on each other. Therefore, an element in one model that ends at a location and chains to another element in a different model must analyze the same in both models. The results of running SLIC on several models is as if all View PVs had been merged together. The detail page created for a layup may have elements copied from separate models. This multiple model capability allows unlimited part size. View PV may be input to any CADAM view scale. SLIC reads the full size values of the geometry.

Line Types

Elements defining layups must be solid or dashed line types. Phantom, Center, NC, or Break line types will be ignored by SLIC for layup definitions.

Arrowheads on Elements

When defining the edge of a layup, an arrow is used with text information. The end of the arrowhead must lie within .004 of the element defining the layup. To

accomplish this, the terminal operator selects /ARW/ under Function Key MISC. To put the arrowhead on the element, the terminal operator selects the element and indicates. the second indicate will establish the tail. The text is then typed in, followed by a third indicate for the positioning of the text. See Figure 16, reference number 161.

Arrowheads on End Points

Do not place identifying arrowheads on the end point of an element. SLIC gives an arrowhead error if placed on end points. See Figure 16, reference number 162.

Layup Edges Must Be "ARROW/TEXT" Defined

The text defining layups must be created under ·ISC.Function/ARW/.arrowhead text. SLIC will not read ındard text at the end of the arrows.

Valid Layup Callouts in "ARROW/TEXT"

The examples in Figure 16, reference number 163, identify the element as belonging to layups 1 through 5.

Unique Layups

Layup numbers may be duplicated in the ply table, provided the duplicate layup number resides in a different sequence. For example, there can be only one geometric boundary definition for L1. SLIC provides no method for distinguishing different boundaries for the same L number. In Figure 16, three identical layups (L97, L98 & L99) are shown (see reference number 164). Even though their internal geometry and material are the same, each must have a separate L number. Conversely, S10 and S40 both use L1.

Relief Cuts

Relief cuts (darts) will be arrow defined as "C#" as opposed to "L#." Each element of a cut must be arrow defined under this rule. See Figure 17. This capability has not yet been implemented.

No Duplicate or Overlapping Elements

Multiple elements will cause the program to identify chaining errors. All elements must be relimited end to end and not be overlapped. Also, two identically duplicated elements will cause SLIC to report chaining errors. See Figure 18.

Gaps Between Elements

Elements must connect with .004, or a chaining error will be identified by SLIC. See Figure 18. This option can be changed for each installation.

Ply Table

The ply table is one of the most important parts of SLIC input. For accurate operation, this table must follow specific rules. Figure 19 shows the ply table. The ply table is supplied and available in the CADAM Standard Library (STDLIB). The ply tables may reside as details of the model. It is recommended that all editing be accomplished in the CADAM Detail Function and be displayed as a DITTO(s).

Ply Table Drawing Sheet

The ply table may reside on any sheet or multiple sheets of a drawing, but <u>must</u> reside in CADAM AUX View 98 of that model. View 98 will not be used for any other purpose throughout the CADAM data base. Any models not requiring the SLIC ply tables will also not use View 98. The view used by your installation for ply tables can be modified in SLIC.

Part Numbers

Each part number (dash number of a drawing) must have its own ply table. This includes opposite parts. Geometry must be noted as opposite if the geometry is to be flipped. Any change to the ply table requires that the

dash number be changed or that a new and additional ply table be generated appropriate to the change. See Figure 19, reference number 191.

Ply Table Entries

The terminal operator may input notes to the ply table at the locations provided. Do not move the start location of the notes. Any individual note within a location must be continuous. The CADAM "$" may be used within the note where required. Commas and dashes are used as layup delimiters similar to geometry identification, as shown in Figure 19, reference number 192.

Ply Table Part Sketch

A small sketch of the part shall be placed in the location shown at reference number 193. The rosette shall also be shown in this area. The rosette must not reside in View 98 with the table. View 99 is suggested for rosette placement. A note or ditto in View 98 which has a degree sign will be interpreted as a legitimate ply orientation and cause a SLIC error. See Figure 19, reference number 194.

Sequence Numbers

The sequence numbers shall be originally entered in increments of ten (10). This permits additions during interative process without renumbering the entire table. Note in Figure 19 how sequence "S31" has been added without necessity of revising the table. SLIC will place the sequence between S30 and S40 for all output data. A sequence number may not be repeated without causing a SLIC error. The lowest numerical sequence is nearest the tool and progresses away from the tool. See reference number 195, Figure 19.

Layup Numbers

Layup numbers may be repeated if the L# is used in different sequences. However, the number of plies and orientations must be the same if the L# is used more than once. This was noted in the paragraph on unique layups. See Figure 19, reference number 196.

Ply Orientation

This column presents ply orientations within specific layups. For example, S30, L15 and L21 have unlike geometry but have the same material and orientation. Also, sequences S30, L15 and L56 have different rotations and material. SLIC provides for every hundredth of a degree (0.01°) of orientation relative to the rosette. The ply orientation is to be input and read from top to bottom, progressing away from the tool. See Figure 19, reference number 197.

Ply Numbers

Ply numbers are input to this column 198. Ply numbers may not be repeated within a layup but may be repeated if the layup number differs. This column is not read by SLIC.

Notes

This space is provided for a flag note only. This space is not read by SLIC. See Figure 19, reference number 199.

Splice Control

This space is provided for flag notes related to splice control only. This space is not read by SLIC. See Figure 19, reference number 200.

Ply Table Revision

This column is provided to account for changes to the ply table. The revision letter should correspond with

the drawing change letter under which the change was made. This space is not read by SLIC. See Figure 19, reference number 201.

Material Callout

Materials are identified by a four-digit integer, as shown in Figure 19, reference number 202. Each layup must have a material code which has been entered into the SLIC Part Data Manager database. The data about any given material must include the data shown in Figure 20.

Material Definition

Figure 20 shows the input required for full SLIC definition of the material. Materials which have not been entered with all of the required data will cause SLIC to report incorrect or no data. This data must be approved by appropriate technology staff representatives and defined to the SLIC program. Materials may include any composite sheet, tape, cloth, honeycomb, sheet metal, and bonding materials, such as glue or cements, which are to be manufactured by laminating, routing or 2 AXIS machining. Any material not defined will not be accounted for by the SLIC program. Undefined material codes will cause SLIC to generate an error message and stop further processing. For all composites, the data shall be in the cured conditions.

SLIC GEOMETRY LOGIC

Geometry Input Logic

This section explains the logic used by SLIC and the rules to be followed by the user for model preparation. The majority of model preparation is to use good drafting standards. Arrowheads should be placed on elements. Text defining layups is to be ARROW/TEXT, use proper cornering, etc. There are a few logic cases where some additional model preparation is necessary. the following rules are concerned with only one layup. Each layup in the part must follow the same rules. The examples use lines for explana-

tions. All rules apply to lines, circles arcs, splines, offset splines and ellipses.

ARROW/TEXT and Chain Logic

SLIC requires every logical section of chained element to contain an arrowhead identifying the layup. This means that were more than one element chains to an end point, an arrowhead must be placed in the next section of the layup. An example is shown in Figure 21.

For the first arrow only, the top section will be identified by SLIC. To completely define the layup, an arrow must be placed somewhere on the section after the three elements chain at the circled points. SLIC will logically chain at end points. If the layup drops off before the end point of an element, an arrow must be used to identify the drop off. An example is shown in Figure 22.

The top and bottom arrows will identify the top and bottom sections. If the ply is to be the center section, the additional arrows are needed. Where drop-offs occur, the arrowhead location is placed on the section of element to be used. See Figure 23.

It is acceptable to use more than one ARROW/TEXT callout on one element to produce the desired geometry.

Another method to produce the desired geometry would be to use two elements, as shown in Figure 26.

The logic for creating layup geometry breaks elements apart when an end point lies on the element to be broken. For example, in the first location, element 2 will be broken by element 1. At the second location, 3 and 4 cannot break each other because neither line has an end point on the other. To enable layup 1 to branch in the correct direction at location 2, either line 3 or 4 must be broken and relimited back to the other line. Only one of the lines should be done, not both.

Program Logic

This section is an explanation of the process SLIC uses to determine a layup. The explanation is given to help in debugging parts. There are three main steps which SLIC uses to determine where a layup is to be defined. The steps are the same for all layups.

(Step 1) For each layup, SLIC creates chained sections that go with each arrowhead. The section chains until it chains to more than one element or doesn't chain at all.

For the example in Figure 29, there are only two elements. The end points are chained until an end point connects to more than one element. These intersecting elements are also placed in the table. After Step 1, the group would appear as shown.

(Step 2) Every element that has the end point of another element on it is then broken into two elements. The table looks the same, except the bottom line is now three lines. It will be broken at the indicated points shown in Figure 30.

(Step 3) The chaining logic is again executed to build the final table. Now the layup can chain in the middle of the bottom element. The final group of chained elements is considered a layup and will be put on a detail page of the appropriate SLIC constructed model, as shown in Figure 31.

The geometry for the layups is found on detail pages of a SLIC-constructed LAYUP model. Each detail page will contain the geometry defining that layup. Lines and arcs are created exactly the same as they are in the original model. Splines, offset splines, and ellipses are created with some differences.

Spline Creation

Splines

Splines are defined using the original defining points. Only that section of the spline needed for the

layup geometry is created on the detail page. The defining spline points will actually be used for approximately six inches from the drop-off or to the end of the original parent spline. This is to allow the splines to be relimited for manufacturing excess purposes.

Offset Splines

Offset splines cannot be recreated as parent splines. Therefore, an offset must be created to carry the mathematical definition. SLIC will create the parent spline on the NOSHOW page and create the relimited offset spline on the show page.

Ellipses

Ellipses are converted into parent splines using the same CADAM logic for conversion of ellipses under the function key spline.

Flange Angle Splines

Flange angle splines are treated like normal splines. The splines created on the layup models do not carry flange angle data on this version of SLIC. This feature could be added if the need arises.

MODEL ARRANGEMENT

The type of part defined as well as the size of the part may dictate various drawing sheet arrangements. This section addresses some of these arrangement combinations.

Drawing Sheets

SLIC can be executed against one or more sheets of a drawing. A single sheet drawing is depicted in Figure 15. In this example, the primary geometry is in CADAM View PV and the ply table in View 98. Figure 32 shows a multi-sheet drawing where one sheet is for the ply table, a second set is for the finished part, and a third sheet is

for the flat pattern of the part. In this example, SLIC would be executed against sheets 1 and 3 only. Sheet 2 describes the finished part and does not define the layups.

Figure 33 presents a multi-sheet drawing where the geometry is too large for placement on one sheet. To aid in sheet reduction, CADAM View PV may be scaled. Any combination of multi-sheet and CADAM multi-models within a sheet may be intermixed satisfactorily, provided all geometry and ply table rules are strictly adhered to.

Panels

Panels, whether flat or contoured, may be input as shown in Figure 15. Note that manufacturing excess is provided around the outside edges. The amount of excess depends upon the manufacturing method and should be determined by planning during the design development.

Flat Patterns

When flat patterns must be developed to produce parts similar to extrusions or sheet metal flat patterns, certain rules must be followed. The flat patterns must all reside in View PV of their drawing and include excess peripheral material for exterior edge trim. Also, each flat pattern must have a unique ply table. A ply table is not required for the assembly drawing of these layups. As with panels, the view containing the geometry may be scaled. Again, the flat panels do not require dimensioning. Figure 34 presents this concept.

ERRORS

Executing SLIC will result in error reporting by creation of a model which reports arrowhead errors, ply table errors, format chart errors, format chart warnings, and a caution chart. In addition, SLIC will display geometric errors when it constructs layup geometry models.

Upon execution of SLIC, a CADAM model will be generated which will modify the 11th through 14th model

identification characters to read "EROR." When the user calls his model, the display may be as shown in Figure 35. The figure shows the five potential error tables which may be generated. Any combination of the five presentations may display. If the heading is not displayed, errors of that kind do not exist. The time and date when SLIC was executed are updated, and this new model overwrites any previous Error Model with the same number. Upon completion of the job, the message "SLIC PROCESSING COMPLETED" will appear on the error model.

The terminal operator may select any of the five error tables shown while in the DETAIL function. The selection will determine which of the following tables will be presented.

Arrowhead Error Table

If arrowheads are not placed correctly on elements, their x-y location will be placed in the Arrowhead Error Table. If an arrowhead has several layups defined on it, only one of the layups will appear in the chart. This table will reside on a detail page of the error model constructed by the program. The detail page generated is shown in Figure 36 and will be identified as noted.

Ply Table Errors

Selection of ply table errors per Figure 35 will display the table shown in Fiugre 37. A brief description of the errors follows the figure.

Dash 101 Page # Already Defined

The same page number exists for the particular dash (part) number. If the dash number requires more than one ply table, the new table must have a different page number.

Dash 101 Has No Page Number

A dash number has been found without a page number attached.

No Dash Number Found

View 98 does not contain dash number callouts.

Dash 101 Sequence #XXX Already Defined

The sequence number appears more than once in the ply tables for a given dash number. This is not allowed for planning purposes.

Dash 101 Sequence # Does Not Match a Layup

The sequence number was unable to find associated layup numbers in the ply table. The layup note either has not been added or is placed in the ply table incorrectly.

Sequence 3 Is Not Found in a Ply Table

A sequence number was found, but it is not in the bounds of any ply table.

No Sequence Numbers Found

View 98 does not contain any sequence number callouts.

Dash 101 Layup #XXX Does Not Match an Orientation

The layup number was unable to find an orientation in the ply table to assign to it.

Dash 101 Layup #XXX Does Not Match Material

The layup number was unable to find an associated material code in the ply table.

Layup #XXX Not Found in a Ply Table

A layup number was found but is not in the bounds of any ply table.

Layup #XXX Contains More Than 8 Ply Orientations

Manufacturing limitations presently require that a maximum of eight plies be related to a unique layup.

No Layup Numbers Found

View 98 does not contain layup number callouts.

Dash 101 Orientation XX.XX° Does Not Match a Layup

The orientation was unable to find a layup number in the ply table to assign to it.

Orientation XXX.XX Not Found in a Ply Table

The orientation is not identified in a ply table.

Orientation XX.XX Not Within Limits of SLIC

Presentations must be greater than or equal to -89.99° and less than or equal to 90.0°.

No Orientations Found

View 98 does not contain orientation callouts.

Dash 101 Material Code 70XX Does Not Match a Layup

The material code was unable to find a layup in the ply table to assign to it.

Material Code 70XX Not Defined

The material information per section 4.3 has not been input to the system.

Material Code 70XX Not Found in a Ply Table

A material code was found but is not in the bounds of any ply table.

No Material Codes Found

View 98 does not contain material code callouts.

Dash 101 Layup #XXX Already Defined, Same Sequence

A layup is defined more than once within the same sequence.

Dash 101 Sequence 20 Overlapping Plies: L1, L2

Layup 1 and layup 2 overlap each other on sequence 20. This error is not related to the ply table, but will be placed in the ply table error chart.

Unable to Chain

End points on the part geometry did not chain together on all the layups. This error is not related to the ply table, but will be placed in the ply table error chart.

Illegal Cutout

Cutouts were found within or overlapping other cutouts on the same layup. This error is not related to the ply table, but will be placed in the ply table error chart.

Exterior Undefined

Unable to find a profile for a given layup number that would contain all other profiles defined in that layup. This error is not related to the ply table, but will be placed in the ply table error chart.

No Geometry in View PV

Models were flagged for the SLIC run that do not contain geometry in view PV. This error is not related to the ply table, but will be placed in the ply table error chart.

No Arrowheads in View PV

Models were flagged for the SLIC run that do not contain arrowheads in View PV defining the plies. This

error is not related to the ply table, but will be placed in the ply table error chart.

Two Pierce Points Have Same ID: ID + XXXX

Two pierce points have been called out on the CADAM drawing with the same ID, or value. The xy location on each point will be given so one of the values can be changed. This error is not related to the ply table, but will be placed in the ply table error chart.

No Valid Number Found for Pierce Point

A pierce point has been called out on the CADAM drawing without a numeric value attached. This error is not related to the ply table, but will be placed in the ply table error chart.

Caution Chart

Selection of the caution chart will provide a display which warns the user of potential problems.

Dash 101 Layup #XXX Already Defined

A layup number can be defined more than once in a ply table, as long as it occurs in separate sequence numbers. This is only a warning message for the user that did not want to revise a layup number.

Format Chart Errors

If the sequence format model is missing information, error messages will be generated on the Format Error Chart. See discussion of Sequence Format Models in this section for more information on SLIC sequence models.

SEQ # Not Found for Dash 101 Format

The SEQ # note was not found in View 99 of the Sequence Format Model.

PG # Not Found for Dash 101 Format

The PG 3 note was not found in View 99 of the Sequence Format Model.

Sheet # Not Found for Dash 101 Format

The sheet number note was not found in View 99 of the Sequence Format Model.

Rev Not Found for Dash 101 Format

The revision note was not found in View 99 of the Sequence Format Model.

Format Chart Warnings

The Format Warning Chart will have message warning of possible problems.

Format Model Not Found for Dash 101

The sequence option has been selected, but a sequence format model was not found for the given dash number. Sequence models will not be generated for this dash number.

Geometry Errors

When all ARROW/TEXT and ply table errors have been corrected, SLIC will produce one or more CADAM models as required to produce CADAM details, each defining the geometry of a unique layup. These details are full size and include geometry joined from multi-sheets to define the enire layup. A maximum of fifty details or the maximum CADAM model size identified at installation time are used and then a second model of details is created. The parent model identification will be used to identify these models, except that the 11th through 14th characters will be changed to "LYUP" and the 17th and 18th characters will identify the numerical identity of the model where multi-models are required for layup details. Selecting a LYUP model will display the table shown in Figure 38.

This table reports each layup which resides in the model, which detail page it is on, and geometry errors. By displaying a detail which reports an error, it can be noted where the error resides so that it can be corrected on the parent model. Correcting the detail does not correct the parent model. The operator should note all errors, correct them on the parent model, and re-execute SLIC. Corrected and newly created LYUP models will overfile (replace) the previous LYUP models.

These cycles must continue until all errors are corrected. If any errors, whether ARROW/TEXT, ply table or LYUP, exist, the downstream output of SLIC will not be generated.

The LYUP models automatically contain plot data for plotting the error chart.

Sequence Overlap Errors

If L numbers on any given sequence overlap each other, an error will result. See Figure 39.

TECHNOLOGY OUTPUT

SLIC will output a variety of data to support technology applications such as design, stress and weights. These include an output table which reports the sequences, layups and plies in proper order, including certain properties. This information table can report raw material requirements and properties or, with minor modifications of the design model, it can report this data for the finished part. A SLIC function called "PIERCE" permits interactive or batch pierce of the designed part and provides strength data about the part. The batch pierce will allow interface to programs such as NASTRAN.

Output Report

Once all errors have been corrected, as discussed in the Errors portion of this section, SLIC will construct a CADAM model which reports data about the designed part.

This model is identified by revising the 11th through 14th identification characters to read "INFO." This information model is shown in Figure 40.

Weights and Center of Gravity

At this time, SLIC will report the weight and CG of the identified layup numbers. These data are included in the INFO model. Because the basic model may show excess material, the data may not reflect the weight and CG of the finished flat pattern. To obtain the data for an as-finished flat pattern, the user should first copy and file the model in the weights CADAM Group. Next, the excess material lines should be erased and the as-finished part periphery defined with solid lines. Next, the layup ARROW/TEXT for the periphery must point to these elements. Upon executing SLIC, the INFO model will present data about the as-finished part. See Figure 41.

Figure 40 shows the original INFO model from the part as constructed in Figure 15, which included excess peripheral material.

Pierce

SLIC will allow a pierce of the composite layups. There are two methods for pierce input and output. The methods are Interactive and Batch. For Pierce to execute, no errors can exist (see Errors portion of this section.)

Interactive Pierce

For interactive pierce, ARROW/TEXT must be used. Figure 42 shows several pierce points on a drawing. These must be input on CADAM View PV and use ARROW/TEXT. Also, the text must be explicit. It must be keyed in "PIERCE POINT." The text contains the pierce point number followed by an optional local rotation. For example: "Pierce Point" will create properties of the laminate using the rotation values exactly as they are in the ply table. "Pierce Point 2,11" will create properties using the values

in the ply table minus 11 degrees. This enables a user to calculate properties at various load direction angles on the laminate. A pierce point number may not be repeated.

To operate this SLIC capability, the CADAM model must filed with the pierce points and SLIC re-executed. This will produce a CADAM model where the 11th through 14th characters of the model identification are changed to "PRCE." This model will construct CADAM details of the pierce data. Upon selecting the PRCE model, the display in Figure 43 will be presented. This model contains the plot data necessary to plot all the pierce charts.

While in the detail function, the pierce point desired for review can be selected, and SLIC will display the appropriate data for that pierce point, as shown in Figure 44.

The data presented by SLIC Pierce is totally dependent upon a model which has all errors corrected as discussed in this section and also upon approved input of material data per the Material Callout portion of this section.

Batch Pierce

The batch pierce option is available for generation of input cards to NASTRAN. This option is run batch from a TSO terminal. The user is required to supply:

1) CADAM group and user.
2) A list of models comprising the part.
3) A list of pierce points.

The models should be verified that they are SLIC compatible.

The pierce points should be in the same coordinate system as the models. They are in the format shown in Figure 45. The points are read in free field. If the local rotation is zero, it needs to be input as 0.0. The point ID is limited to 6 numeric digits.

Job Control Language (JCL), as illustrated in Figure 46, is submitted.

The input points are in file FT11F001. The diagnostic messages appear in file FT09F001. The NASTRAN input cards appear in file FT10F001.

PLEFT Option

This is an option in which all angles within the layup are multiplied by -1 before the properties are computed. This was done so that laminate properties could be taken directly from the SLIC model and input to a NASTRAN Finite Element Model.

Section Cuts

The SLIC program has the capability to generate section cuts through properly defined parts. The section cuts models produced contain both stick figures, plots of thickness, ply count, and moduli of the desired cut. In order to obtain this information option, attribute under FK/Group needs to be utilized to define the location of the cut. A line needs to be defined in the location where the desired section cut is desired; this can be on the same model as the part or a separate model. If a new model is created for this, the PV origins must be the same in both models. When assigning attribute data, three requirements exist:

1) Attribute number 650 is assigned to all section cut lines.
2) Part dash numbers must be assigned to all section cut attributes.
3) User-defined identification numbers must be assigned to all section cut attributes.

The dash number and section cut identification number are input as attribute data and should be input per the CADAM user's manual. The only requirement on this attribute data is that it be put in as follows: "n,m."

n = integer defining unique section cut line.
m = integer defining part dash number.

The direction (or perspective) of the section cut obeys the right-hand rule. A line cannot be used for a section line and geometry definition.

Upon execution of the section cut option, two models will be created for each cut line. One model will contain gauge data as in Figure 47. The other model will contain E and E*T plots.

SEQUENCE DRAWINGS

Sequence Format Models

Sequence drawings are presented in the size furnished by the Sequence Format Model. The user creates the model which is copied into each SLIC-created sequence models when SLIC runs, as shown, for example, in Figure 48.

The Start Format for the Sequence Format shall be obtained from the STDLIB. This format must be used with some explicit rules. The View PV Origin must correspond with the Origin of View PV of the parent model. For this document example, Figure 15 shows this orientation.

The user shall then copy and transfer any peripheral geometry to View PV of the Sequence Format Model, View Scale PV, and/or the Format View, and move as required to obtain the proper orientation of the format to the picture. Add any location identifiers and change all picture lines to phantom or center, as desired. Any notes which are constant across all Sequence models may be added to this format.

The Drawing Number, Sheet Number, and Revision Letter shall be edited. The Sequence number shall be NO-SHOwn. The page number shall be edited to contain the page number of the first Sequence Page. This note is then "NO-SHOwn." Do not erase these two entries as they are required by SLIC. The rosette shall be added where the user determines clarity is best served. Plot data should be added to this model at this time so that all picture models constructed by SLIC will include this data. See Figure 49 for a Prepared Sequence Format Model.

Picture Format Model ID

The sequence format model will be filed by the operator. This model will be retrieved in SLIC when sequences are run. The user does not include this model in his run. SLIC will look for the model and run the sequences for the dash numbers for which a sequence format model exists. Because SLIC will look for the model, it is important that the sequence picture page model name be exact. The name will be as follows:

1) The first through tenth characters shall contain the same characters as the parent model.
2) The eleventh through fourteenth characters will be revised to read "SEQ0" (last character is zero).
3) The fifteenth through eighteenth characters will be revised to designate dash number. The dash will always reside in the fifteenth character. A single digit dash number will have the number in the eighteenth character, leaving the sixteenth and seventeenth characters blank.

   Example: 1234567890SEQ0,- , 1,--

   SLIC would generate models:

   1234567890S002,03,01,--
   1234567890S003,03,01,--
   1234567890S004,03,01,-- etc.

Sequence Output

Upon executing Sequences, SLIC will construct Sequence page 2 and on, as shown in Figure 50. These pages should be plotted and checked for errors. If geometry errors exist, they should be corrected on the parent sheets (geometry ply tables and formats) and Sequences re-executed.

Each layup of the sequence is presented by CADAM heavy lines as a CADAM DITTO. The layup will include:

Layup Numbers	L1
Each Ply and Orientation	P1 = 90.0
	P2 = 0.0

Due to space limitations, the data from one layup may overlie the data for another layup. The data may be moved on the detail page which represents the layup. If all sequence models do not get created, check CADAM drawfile size.

SLIC generates additional information on the NOSHOW page of the sequences for traceability.

EXECUTING SLIC

The method an installation determines to run SLIC can vary. In order for SLIC to run the model IDs, group and user must be supplied to the JCL that runs SLIC. The following is a recommended method that allows a user the capability to run SLIC from a CADAM terminal and get results back in a short time period. This method requires a job to be set up in the IJPTABLE for submittal of SLIC.

In this case, three options exist from the terminal. The following are the required steps:

- Access the user listing where the models reside.
- Select /2/ on the menu.
- Key in a VOLID. The operator may key in up to six characters to identify the job. Each operator should use caution not to repeat a VOLID to assure that two active tasks on the computer do not have the same ID. Once a job is complete on the system, a particular VOLID may be reused.
- Select those CADAM models against which SLIC is to be executed. These models must be for a particular drawing set. One should not attempt to execute SLIC against unassociated models under a single VOLID.
- Press the Y/N Key when all models have been selected. This removes the M on the left and actually places the VOLID on the models.

- Select /MENU 1/.
- Select /RETURN/ to access a CADAM model.
- Select /DATA-M/.
- Select /IJP/.
- Display will provide three SLIC options. Select the desired SLIC program from the batch job table.

  SLIC - WITHOUT PLY TABLE

  SLIC - WITH PLY TABLE, PIERCE

  SLIC - SEQUENCES
- Select /SCHEDULE/ on the menu.
- Key in the VOLID placed on the models.
- Key in the name of the operator submitting the job.
- Press the Y/N Function Key to begin execution.
- Select /MENU 1/ to exit.

The operator may then proceed to other CADAM tasks and periodically check the user listing for the completion of the job. Depending on the size of the SLIC task, the completion of the job can take from a mere instant up to several minutes.

"SLIC - WITHOUT PLY TABLE"

This option will create models which describe ARROW/TEXT errors, and when no errors exist, models of each layup geometry entity for correction of geometric errors. This output is described in the Errors portions of this section.

"SLIC - WITH PLY TABLE, PIERCE"

This option executes all capabilities described in the Errors and Technology Output portions of this section.

"SLIC - SEQUENCES"

This option executes the capabilities described in the Sequence Drawings portion of this section.

Section Cuts

This option executes all capabilities described in the Technology Output portion of this section.

Although the present invention has been described herein primarily in terms of its embodiment in SLIC, it should be understood that the invention is not limited to this preferred embodiment, but rather includes all equivalent embodiments.

Claims

1. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means communicating with the isolation and definition means for generating tool path data for a numerically controlled cutting machine or the like for cutting the individual plies from a sheet of ply material;

2. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means for allowing pierce point interrogation of the composite part to determine the structural properties at selected points on the part.

3. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means for interfacing with a finite element modeler to allow structural analysis of the complete part.

4. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means for calculating the weight and centroid of the composite part.

5. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means for selecting a desired cross section through a part and generating ply layer and thickness plots for the selected cross section.

6. A system to aid in the design and manufacture of composite parts, the composite parts fabricated from a plurality of plies of material assembled and cured in a desired orientation and arrangement to form the part, the system comprising:

a central processing unit;

input means communicating with the central processing unit;

storage means communicating with the central processing unit;

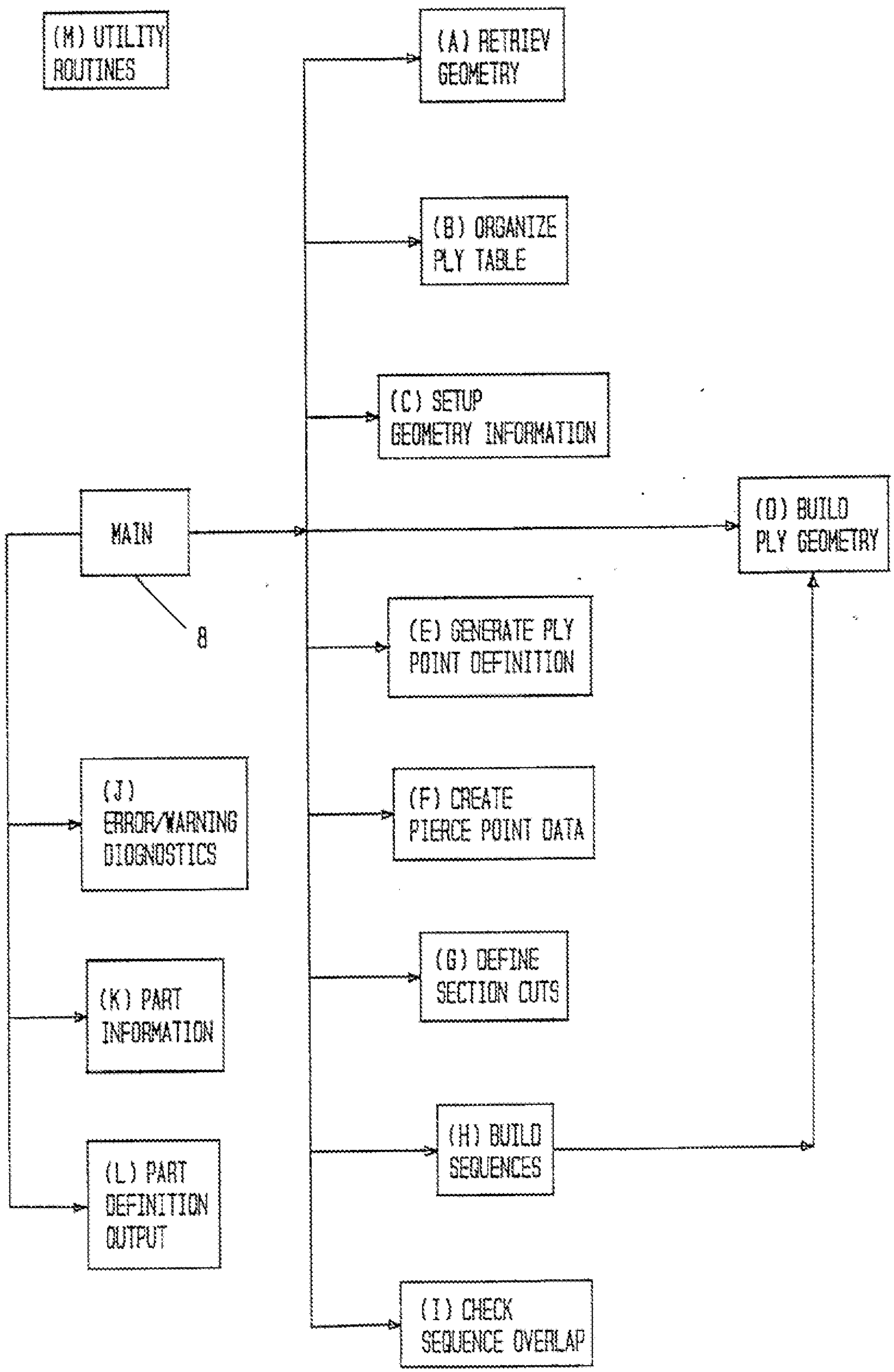
means for inputting and storing information regarding the shape, orientation and location of the plies within the composite part;

means for isolating and defining individual plies within the part; and

means for interfacing with a nesting system to enable ply data to be automatically used to calculate preferred nesting arrangements to efficiently lay out individual plies shapes for cutting in the desired orientation from a sheet of ply material.

MAIN FLOW

FIG.1



(A)

RETRIEVE GEOMETRY

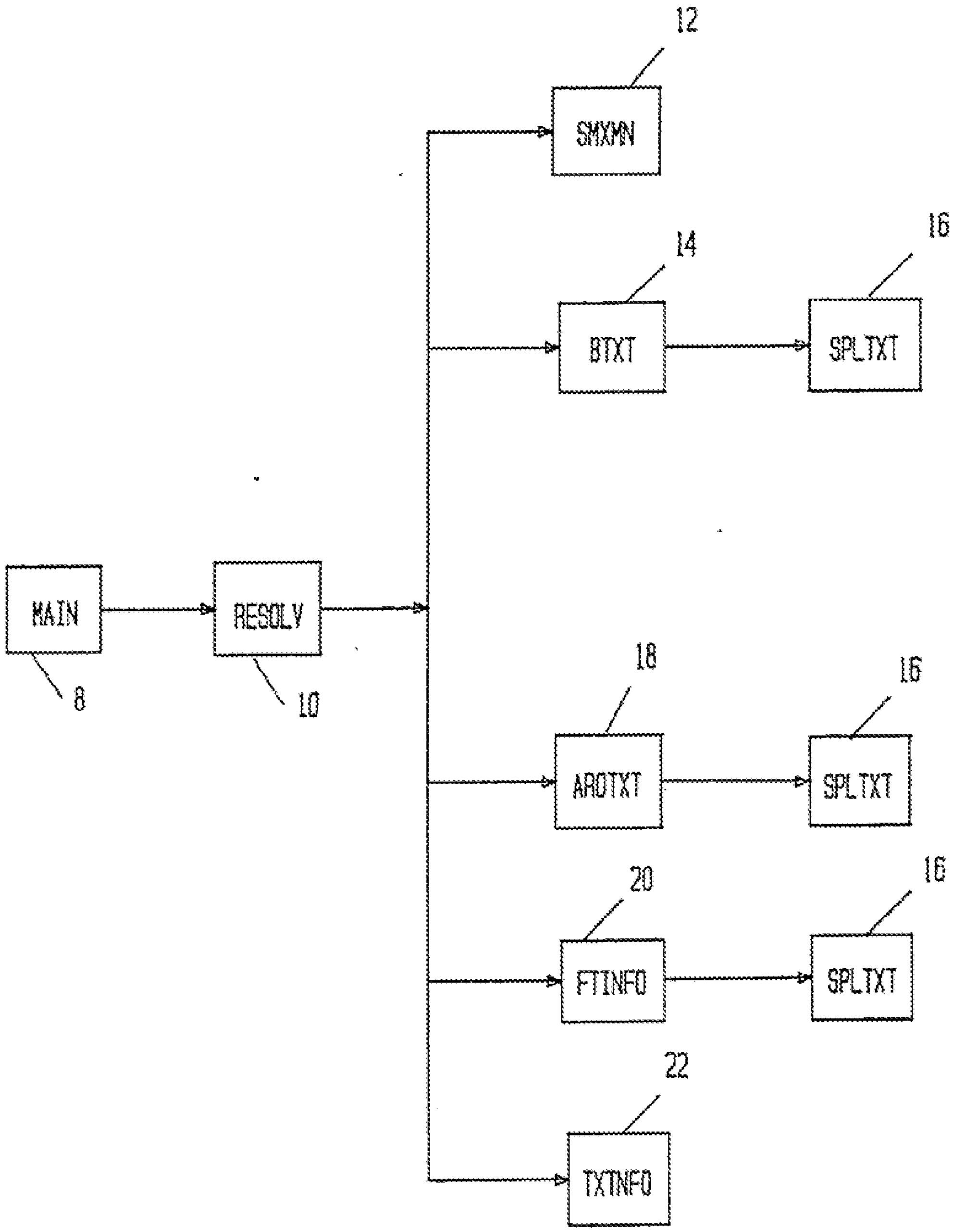
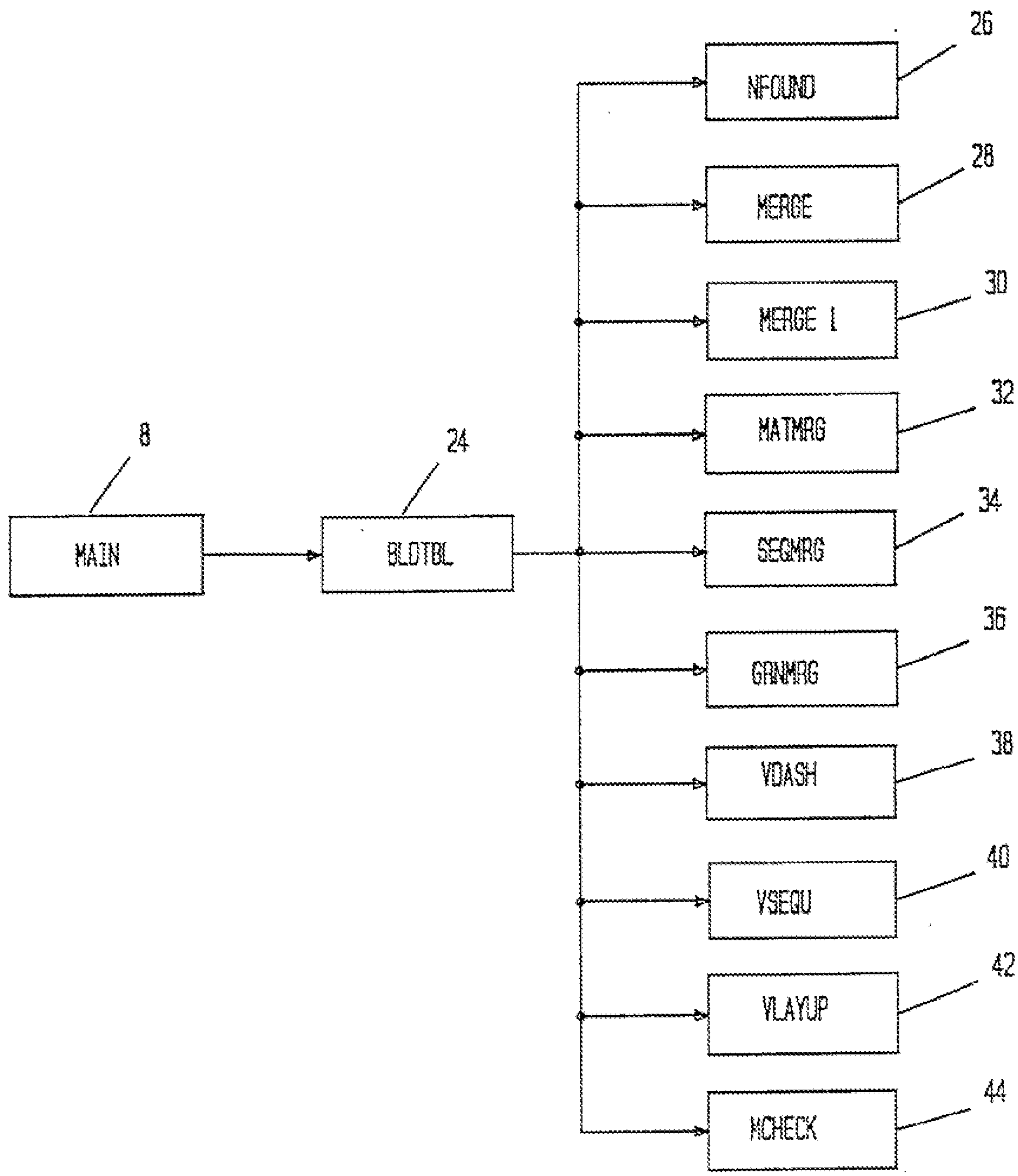


FIG.2

ORGANIZE PLY TABLE

FIG.3



SET UP GEOMETRY INFORMATION

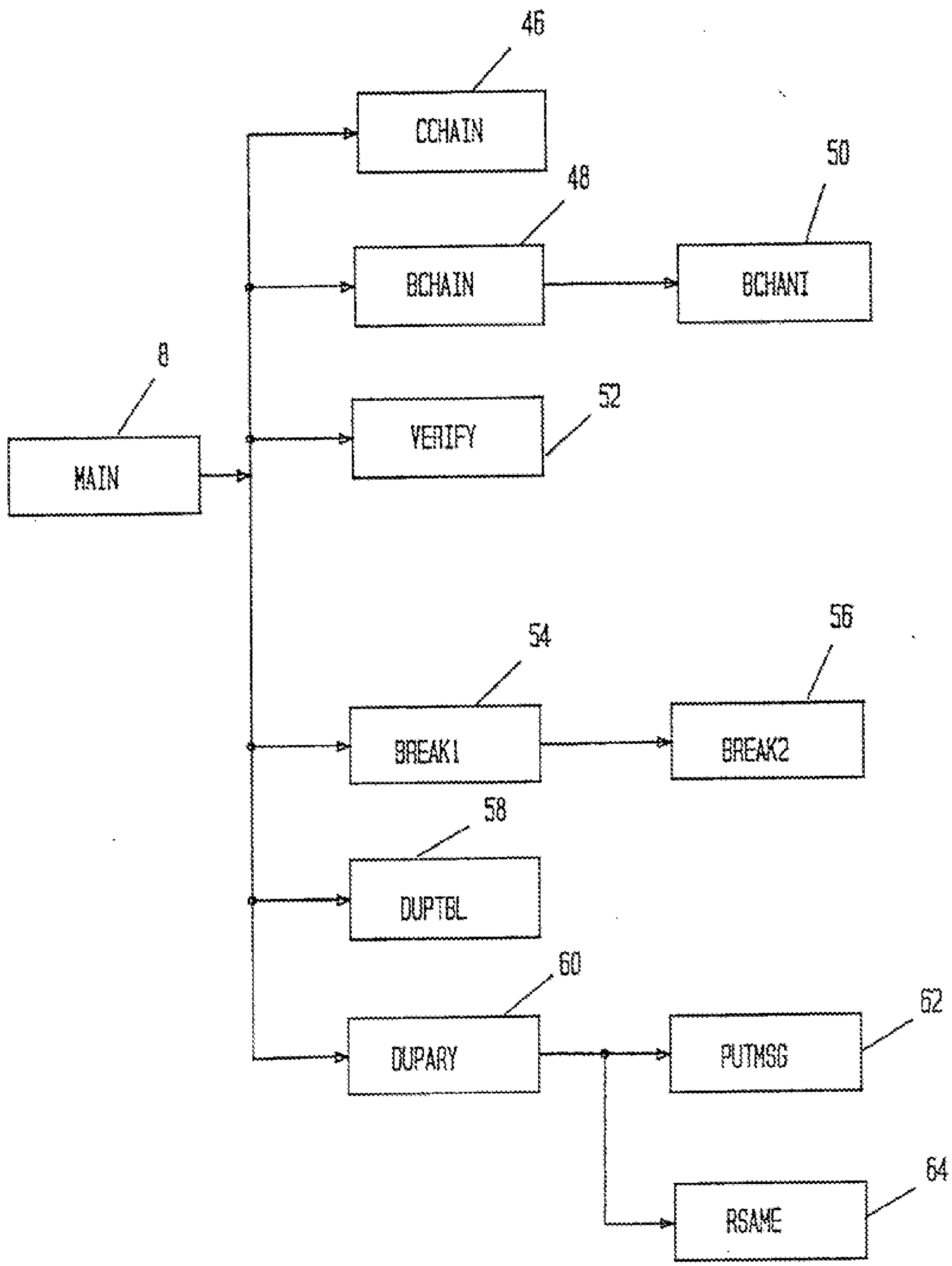
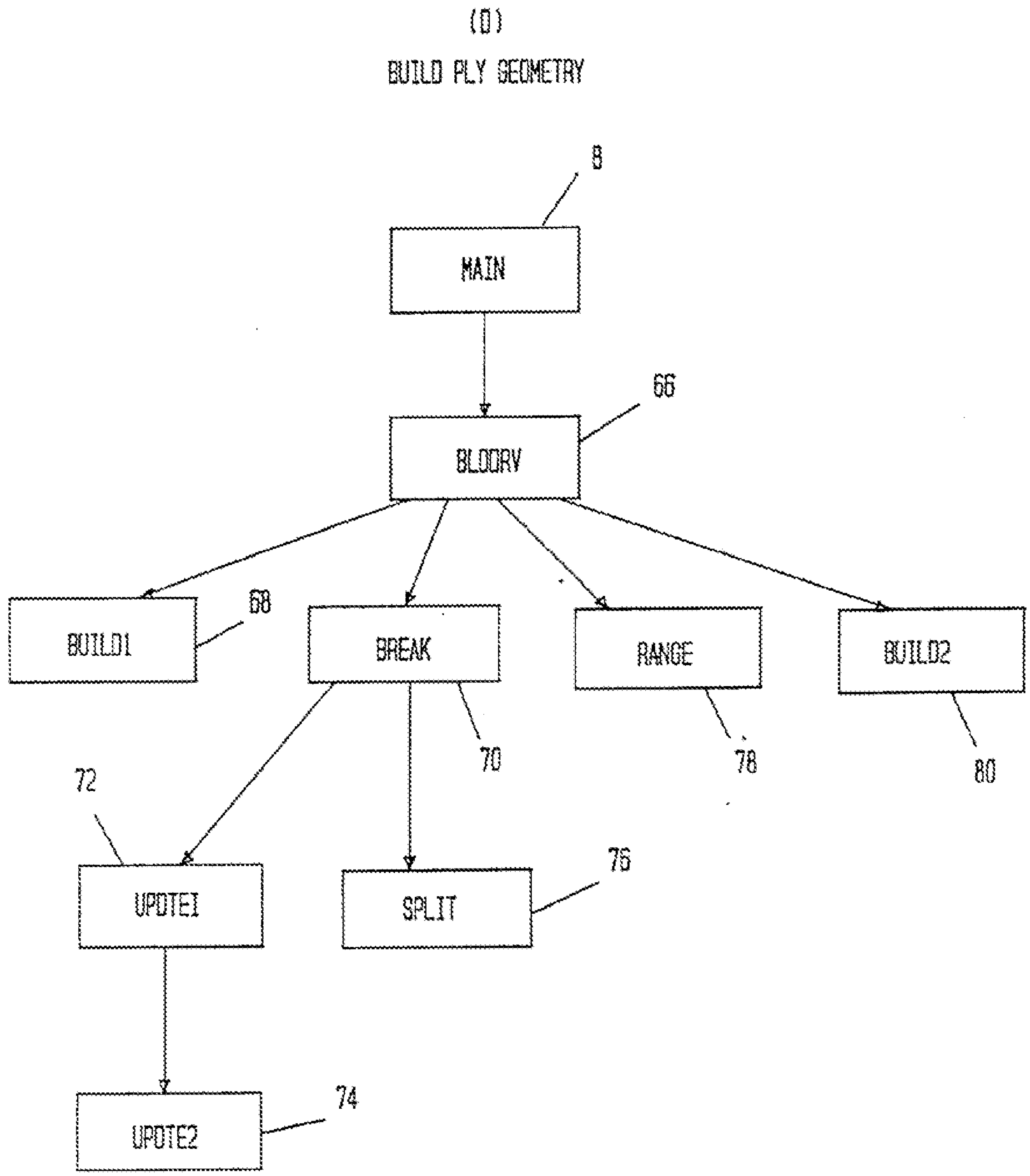


FIG.4

FIG.5A


(D)
BUILD PLY GEOMETRY
8
MAIN
66
BLDDRV
BUILD1
68
BREAK
RANGE
BUILD2
70
78
80
72
UPDTE1
SPLIT
76
UPDTE2
74

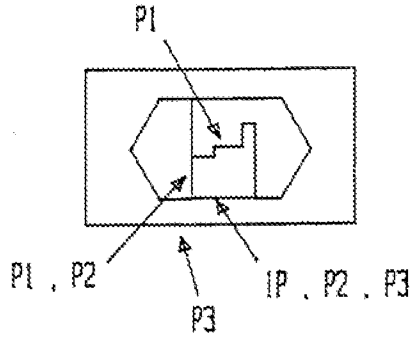


FIG.5B

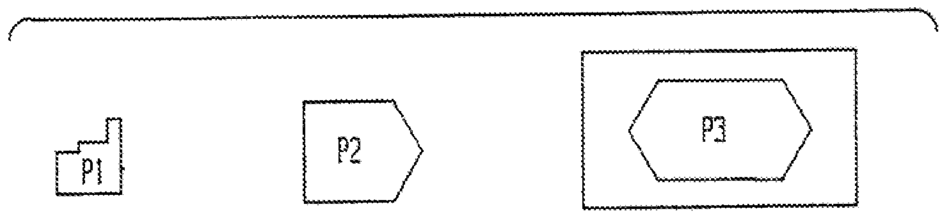


FIG.5C

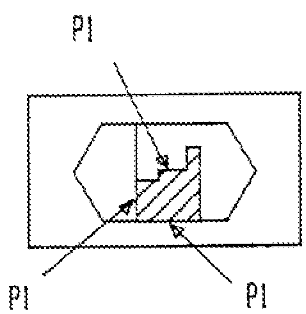


FIG.5D

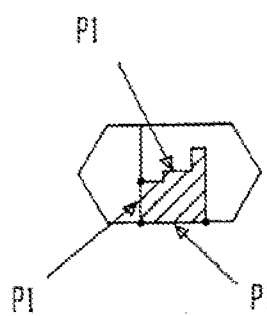


FIG.5E

FIG.5F

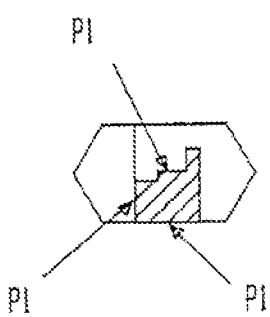
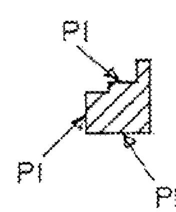


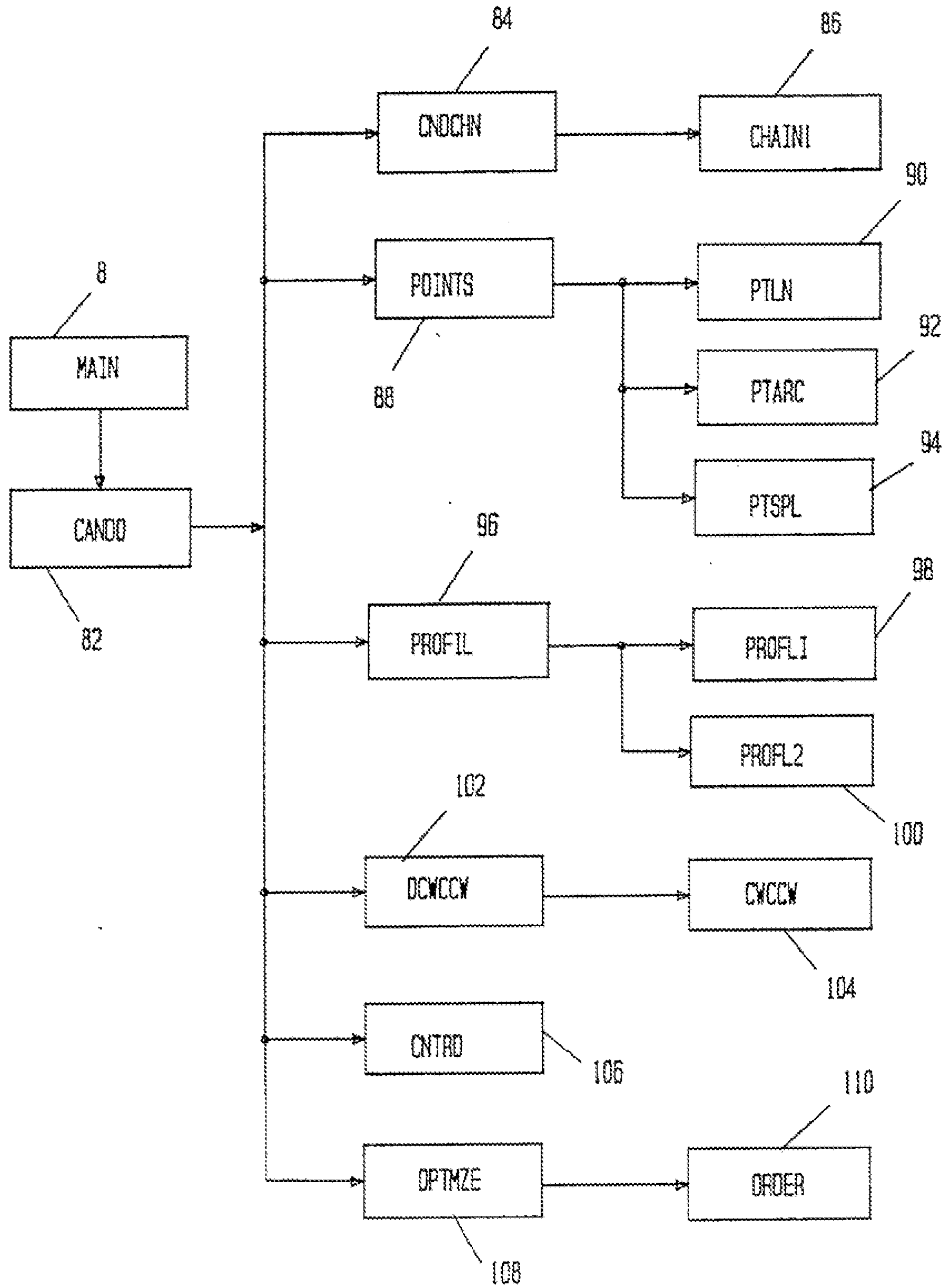
FIG.5G



(E)

FIG.6

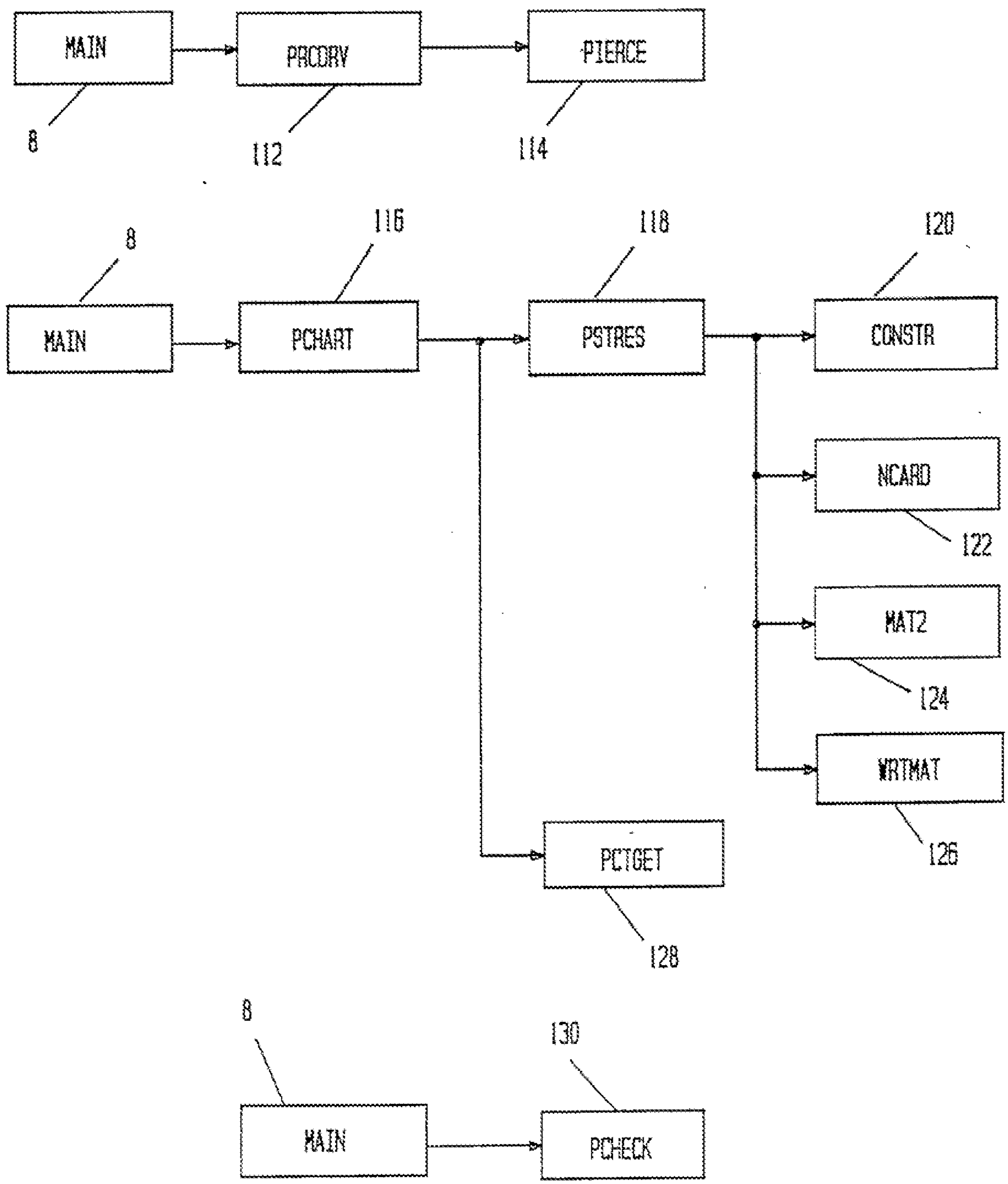
GENERATE PLY POINT DEFINITION



(F)

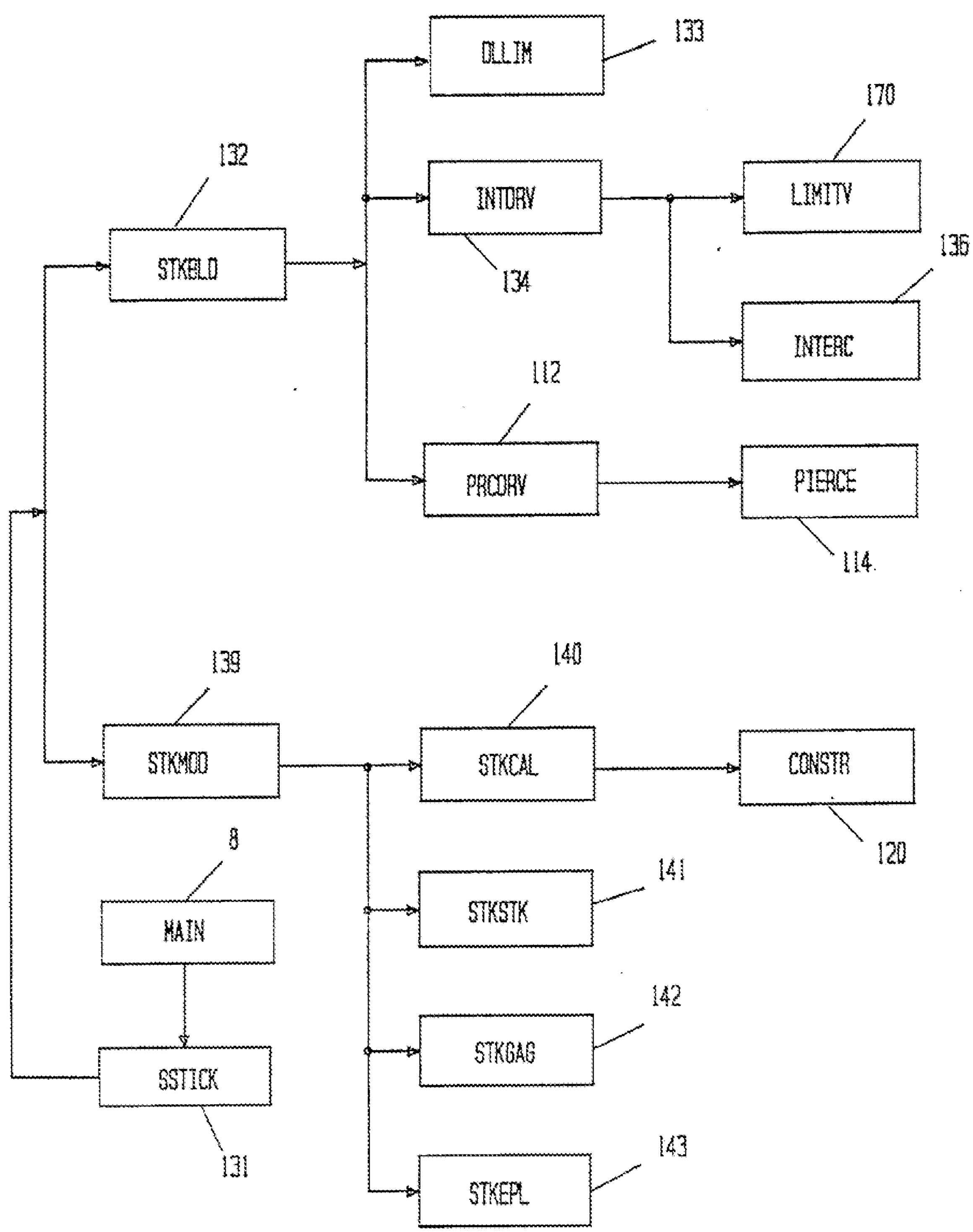
CREATE PIERCE POINT DEFINITION

FIG.7



(G)
DEFINE SECTION CUTS

FIG.8



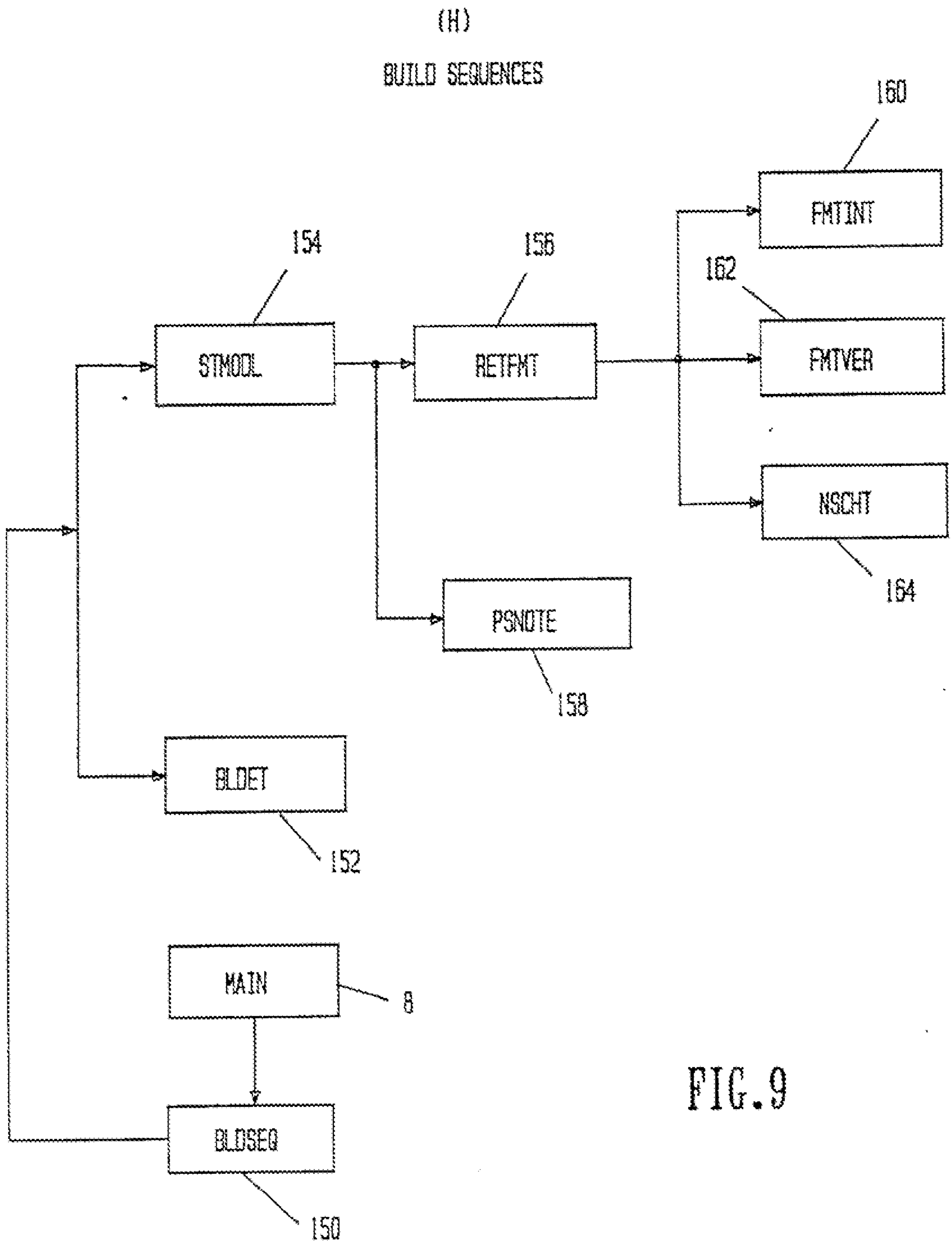
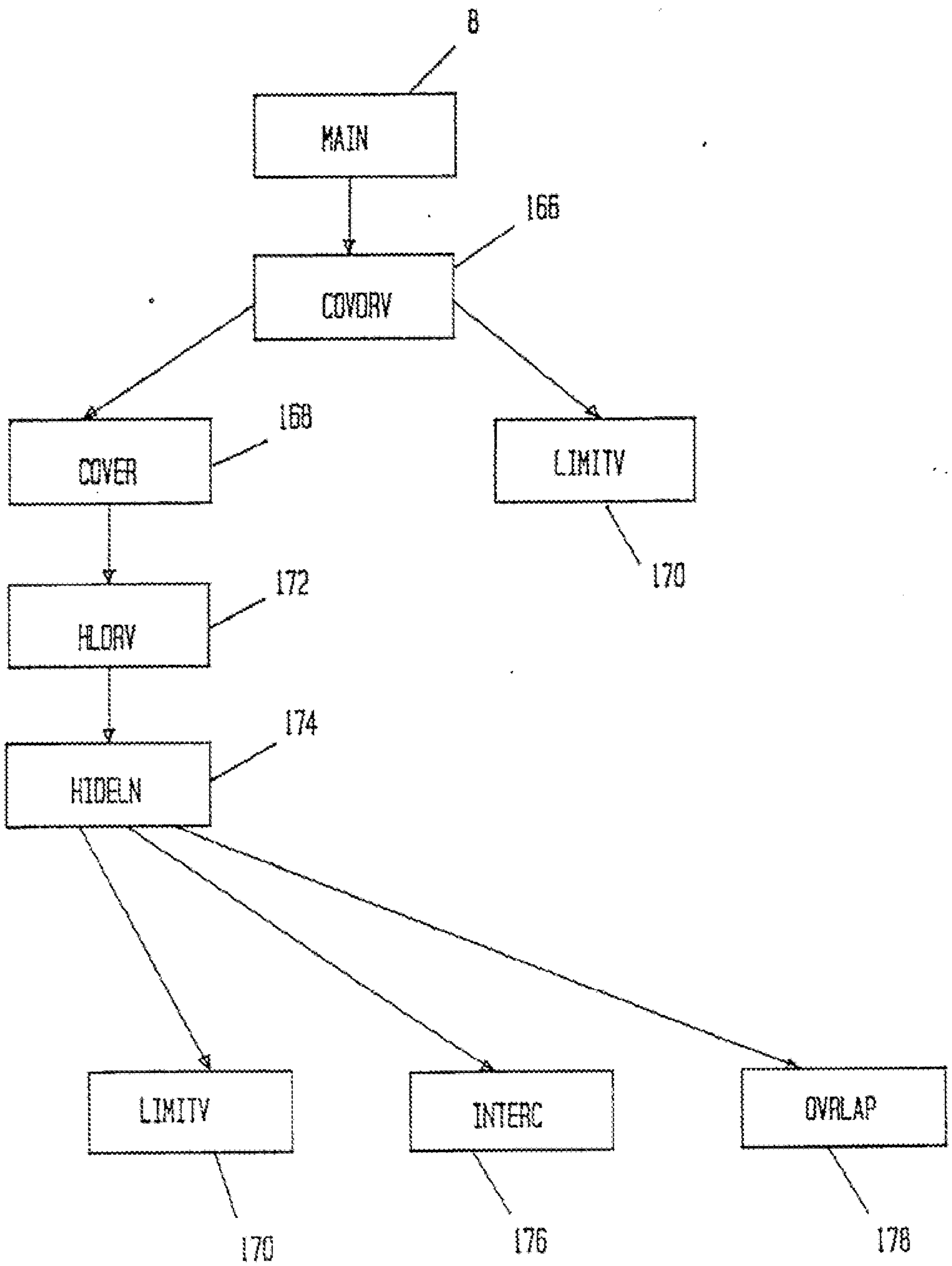

(H)
BUILD SEQUENCES
160
FMTINT
154
156
162
STMODL
RETFMT
FMTVER
NSCHT
164
PSNOTE
158
BLDET
152
MAIN
8
BLDSEQ
150

FIG.9

(I)

FIG.10

SEQUENCE OVERLAP



(J)

ERROR DIOGNOSTICS

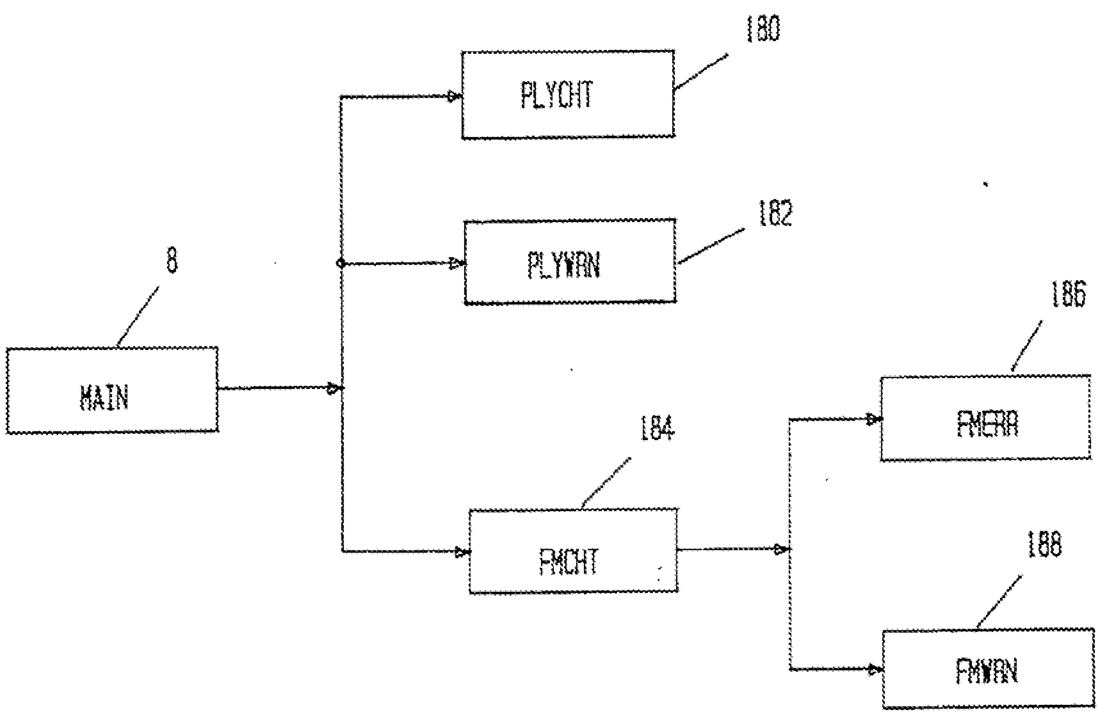


FIG.11

FIG.12

(K)

PART INFORMATION

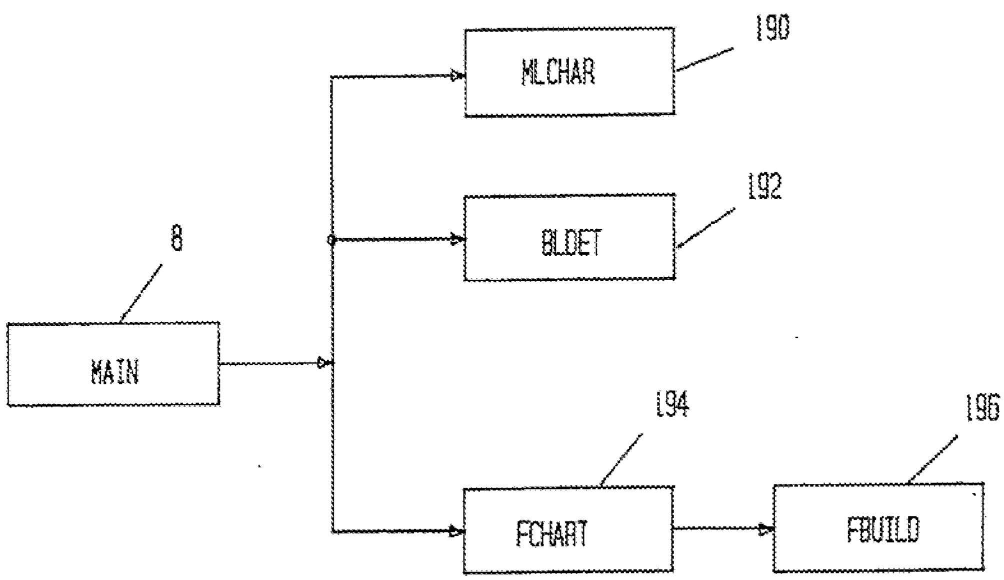


FIG.13

(L)

PART DEFINITION OUTPUT